

МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ

Учреждение образования  
«Гомельский государственный университет  
имени Франциска Скорины»

Факультет физики и информационных технологий  
Кафедра теоретической физики

Допущена к защите  
Зав. кафедрой Тюменков Г.Ю.

" \_\_\_\_ " \_\_\_\_\_ 20 \_\_\_\_ г.

**Анализ особенностей периферических  
взаимодействий легких релятивистских ядер в  
ядерной эмульсии**

Магистерская диссертация

Исполнитель  
студентка группы КФ-57

О.Н. Кашанская

Научный руководитель  
заведующий кафедрой теоретической физики,  
к.ф.-м. наук, доцент

Г.Ю. Тюменков

Научный консультант  
Рецензент  
профессор кафедры оптики  
д.ф.-м.н., доцент

С.А. Хахомов

Гомель 2023

## РЕФЕРАТ

Дипломная работа 76 страниц, 35 рисунков, 12 таблиц, 14 источника, 10 приложений.

**Ключевые слова:** ядерная фотоэмульсия, кластеры, фрагментация ядер.

**Объект исследования:** изучение явления фрагментации ядер во взаимодействиях ядер Be-9 и B-10 высоких энергий с различными ядрами мишени из состава ядерной фотоэмульсии.

**Цель работы:** сравнительный анализ данных эксперимента БЕККЕРЕЛЬ на нуклотроне ОИЯИ по релятивистской фрагментации ядер  $^{10}\text{B}$  и  $^9\text{Be}$  в ядерной эмульсии для выявления особенностей образования фрагментов ядер мишеней различной ионизирующей способности.

**Методы исследования:** методика ядерных фотоэмульсий.

**Результаты:** освоена методика ядерных фотоэмульсий; дополнена существующая статистика событий; проведен сравнительный анализ экспериментального материала по топологии фрагментации ядер-мишеней из состава ядерной эмульсии под действием релятивистских ядер Be-9 и B-10; изучены корреляционные зависимости множественности  $b$ ,  $g$  и  $s$  - частиц во фрагментации ядер-мишени из состава ядерной фотоэмульсии под действием релятивистских ядер  $^{10}\text{B}$  и  $^9\text{Be}$ ; дан сравнительный анализ по средней множественности частиц в зависимости от типа ядра-мишени.

**Область применения:** в периферических взаимодействиях релятивистских ядер  $^{10}\text{B}$  и  $^9\text{Be}$  в ядерной эмульсии исследуется взаимосвязь кластерных и нуклонных аспектов в структуре этих ядер. Кластерная структура легких ядер является ключевым аспектом в широком разнообразии реакций и может оказаться источником для поисков разреженных ядерных состояний. Учет кластерного характера ядерной фрагментации важен при исследованиях по радиационной медицине и геологии.

Исследование кластеризации в релятивистском подходе может дать особые преимущества, поскольку в условиях очень малых передач энергии–импульса в конечных состояниях фрагментов должна наиболее полно отражаться структура начальных состояний ядер.

**Практическая значимость работы:** современные эксперименты проводятся с релятивистскими радиоактивными ядрами с большим избытком нейтронов на магнитных спектрометрах. Они ориентированы на регистрацию фрагментов с начальным зарядом как у исследуемого ядра или близким к нему. Однако в таком подходе теряются принципиально важные каналы, содержащие фрагменты  $\text{He}$  и  $\text{H}$ , и, соответственно, распады  $^8\text{Be}$  и  $^9\text{B}$ . Возможность разрешения этой проблемы методом ядерной эмульсии мотивировала эксперимент БЕККЕРЕЛЬ на нуклотроне ОИЯИ, частью которого является и данная работа.

## СОДЕРЖАНИЕ

РЕФЕРАТ .....	2
СОДЕРЖАНИЕ .....	3
ТЕРМИНЫ И ОПРЕДЕЛЕНИЯ.....	5
ОБОЗНАЧЕНИЯ И СОКРАЩЕНИЯ .....	6
ВВЕДЕНИЕ .....	7
1 Применение метода ядерной эмульсии в физике релятивистских ядер.....	11
1.1 Развитие исследований фрагментации релятивистских ядер.....	11
1.2 Возможности метода.....	12
1.3 Особенности релятивистской фрагментации ${}^6\text{Li}$ , ${}^{10}\text{B}$ , ${}^{14}\text{N}$ .....	14
1.4 Метод ядерных фотоэмульсий.....	15
1.5 Общая характеристика ядерной фотоэмульсии .....	17
1.6 Механизм образования следов.....	20
2 Топология фрагментации налетающего ядра.....	22
2.1 Фрагментация ${}^9\text{Be} \rightarrow 2\text{He}$ .....	22
2.2 Фрагментация ${}^{10}\text{B} \rightarrow 2\text{He}(+H)$ .....	25
2.3 Сравнение и резюме.....	27
3 Фрагментация ядер мишени под действием релятивистских легких ядер ..	29
3.1 Особенности фрагментации ядер мишени в событиях ${}^9\text{Be} + Et$ и ${}^{10}\text{B} + Et$ .....	29
3.2 Представление результатов с помощью ROOT .....	39
ЗАКЛЮЧЕНИЕ .....	43
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ.....	44
ПРИЛОЖЕНИЯ.....	45
ПРИЛОЖЕНИЕ А .....	45
Матрица распределений по множественности для канала ${}^9\text{Be} \rightarrow 2\text{He}$ .....	45
ПРИЛОЖЕНИЕ Б .....	47
Матрица распределений по множественности для канала ${}^{10}\text{B} \rightarrow 2\text{He}$ .....	47
ПРИЛОЖЕНИЕ В.....	48
Матрица распределений по множественности для канала ${}^{10}\text{B} \rightarrow 2\text{He} + H$ .....	48
ПРИЛОЖЕНИЕ Г .....	49
Программный код для построения нормированного распределения по множественности nh, nb, ng, ns ( ${}^9\text{Be}$ ), где $nh \geq 0$ .....	49

ПРИЛОЖЕНИЕ Д .....	51
Программный код для построения нормированного распределения по множественности $nh$ , $nb$ , $ng$ , $ns$ для канала $^{10}B \rightarrow 2He + H$ , где $nh \geq 0$ .....	51
ПРИЛОЖЕНИЕ Е.....	53
Программный код для построения нормированного распределения по множественности $nh$ , $nb$ , $ng$ , $ns$ для канала $^{10}B \rightarrow 2He$ , где $nh \geq 0$ .....	53
ПРИЛОЖЕНИЕ Ж.....	55
Программный код для построения корреляции множественности от $nh$ для канала $^{10}B \rightarrow 2He$ .....	55
ПРИЛОЖЕНИЕ И .....	61
Программный код для построения корреляции множественности от $nb$ для канала $^{10}B \rightarrow 2He$ .....	61
ПРИЛОЖЕНИЕ К.....	67
Программный код для построения корреляции множественности от $ng$ для канала $^{10}B \rightarrow 2He$ .....	67
ПРИЛОЖЕНИЕ Л .....	73
Программный код для построения корреляции множественности от $ns$ для канала $^{10}B \rightarrow 2He$ .....	73
ПРИЛОЖЕНИЕ М.....	78
Заключение о результатах проверки магистерской диссертации на отсутствие плагиата .....	78

## **ТЕРМИНЫ И ОПРЕДЕЛЕНИЯ**

В настоящей диссертационной работе применяют следующие термины с соответствующими определениями:

Кластер — коррелированная группа элементарных частиц.

## ОБОЗНАЧЕНИЯ И СОКРАЩЕНИЯ

μ - микро,  $10^{-6}$ ;

кэВ - килоэлектронвольт;

МэВ - мегаэлектронвольт;

ГэВ - гигаэлектронвольт;

ОИЯИ - Объединенный Институт Ядерных Исследований;

мрад – мили ( $10^{-3}$ ) радиан;

ТВ – терабайт.

## ВВЕДЕНИЕ

Настоящая работа посвящена исследованию периферических взаимодействий релятивистских ядер  $^{10}\text{B}$  и  $^9\text{Be}$  в ядерной фотографической эмульсии для развития представлений о взаимосвязи кластерных и нуклонных аспектов в структуре этих ядер. Работа опирается на возможности нуклотрона ОИЯИ по формированию пучков легких релятивистских ядер, включая радиоактивные, а также многолетний опыт использования методики ядерных эмульсий в Лаборатории физики высоких энергий ОИЯИ имени В. И. Векслера и А. М. Балдина и связанных с ней научных учреждений. Представленные результаты являются реализацией одной из задач эксперимента БЕККЕРЕЛЬ на нуклотроне ОИЯИ эмульсионного сотрудничества БЕККЕРЕЛЬ по изучению кластерных степеней свободы в легких стабильных и радиоактивных ядрах.

Кластеризация нуклонов в структуре легких ядер проявляется в разнообразных реакциях в образовании легчайших ядер  $\text{He}$  и  $\text{H}$ . Общим свойством последних является отсутствие связанных возбужденных состояний. Прежде всего к их числу относится  $\alpha$ -частица, представляющая собой сильно связанную (энергия связи до 28 МэВ) четырехнуклонную корреляцию спаренных по спину протонов и нейтронов с полным нулевым спином. В ядрах с нечетным числом нейтронов или протонов к  $\alpha$ -кластерной основе присоединяются кластеры, которым соответствуют дейтрон (d), тритон (t) или ядро  $^3\text{He}$ , а также протоны (p) и нейтроны (n), связанные вне кластеров (рисунок 1).

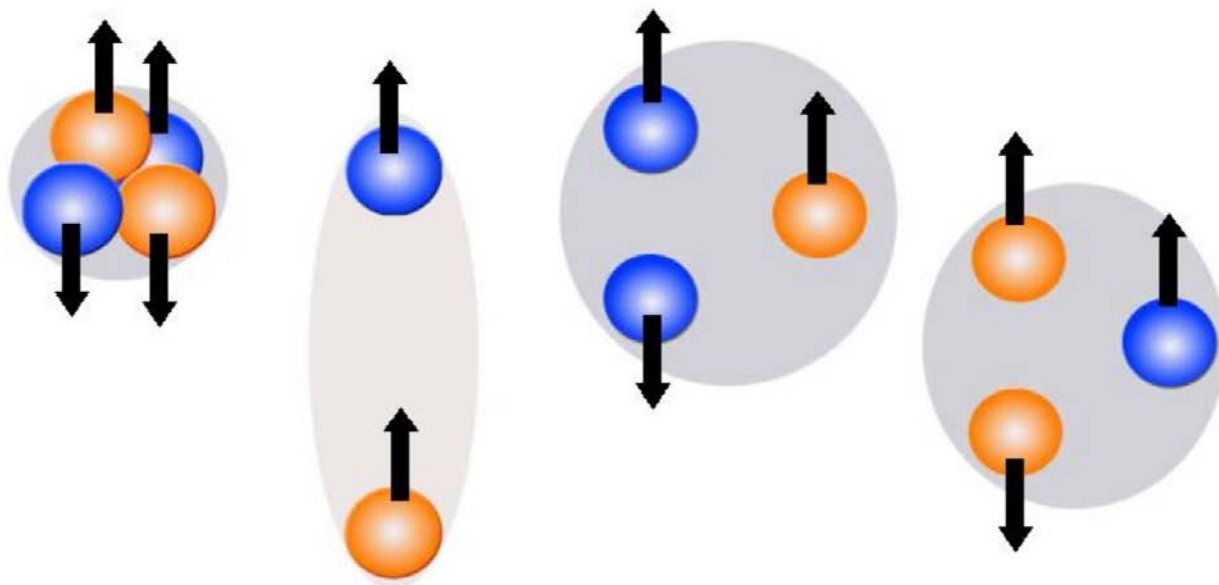
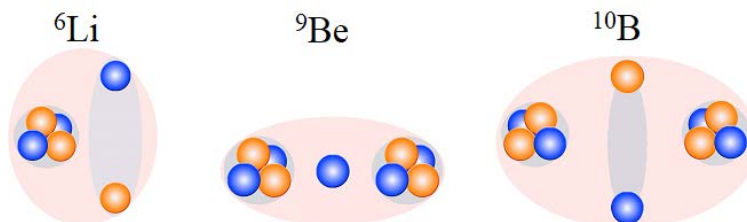


Рисунок 1 – Схематическое изображения легчайших кластеров:  $^4\text{He}$ , дейтрон (d), тритон (t),  $^3\text{He}$ . Синим цветом изображены нейтроны, красным - протоны. стрелки обозначают спин

В таком подходе в эксперименте БЕККЕРЕЛЬ на нуклотроне ОИЯИ изучается кластерная структура легких стабильных и радиоактивных изотопов. Задача данной работы состоит в сравнительном анализе данных эксперимента БЕККЕРЕЛЬ на нуклотроне ОИЯИ по релятивистской

фрагментации ядер  $^{10}\text{B}$  и  $^9\text{Be}$  в ядерной эмульсии для выявления особенностей фрагментации ядер мишени из состава ядерной фотоэмульсии. Детальный анализ фрагментации ядер мишени исследуемых ядер ранее не проводился, оставаясь «в тени» релятивистской фрагментации.

Вслед за основной кластерной конфигурацией  $\alpha d$  в изотопе  $^6\text{Li}$  структура  $^9\text{Be}$ ,  $^{10}\text{B}$  и  $^{14}\text{N}$  представляется как  $2\alpha$ - или  $3\alpha$ -частичная основа со слабо связанным дейтроном или же протоном и нейтроном (рисунок 2).



**Рисунок 2 – Кластерная структура ядер  $^6\text{Li}$ ,  $^9\text{Be}$ ,  $^{10}\text{B}$**

Ядро  $^8\text{Be}$  с ярко выраженной структурой двух  $\alpha$ -частиц имеет крайне малое время жизни ( $\sim 10^{-16}$  с). Присоединение к нему протона ведет к еще более короткоживущему ядру  $^9\text{B}$  ( $\sim 10^{-18}$  с), а нейтрона – к стабильному изотопу  $^9\text{Be}$ . Разительное отличие в зеркальной паре ядер указывает на принципиальную роль нейтронов. Благодаря захвату  $\alpha$ -частиц из распадов тяжелых изотопов структурой  $2\alpha n$  изотоп  $^9\text{Be}$  служит источником нейтронов, начиная с открытия этой частицы. Захват тепловых нейтронов изотопом  $^{10}\text{B}$  со структурой  $3\alpha d(n\text{p})$  широко применяется в ядерно-физических приложениях.  $^{14}\text{N}$  относится к малому числу нечетно-нечетных стабильных изотопов, включающему  $^2\text{H}$ ,  $^6\text{Li}$ ,  $^{10}\text{B}$  и  $^{50}\text{V}$  (ванадий). Его структура представляется промежуточной между  $3\alpha$ -кластеризацией  $^{12}\text{C}$  с присоединением  $d(n\text{p})$  и дважды магической оболочечной  $^{16}\text{O}$ . Кластеризация изотопа  $^{14}\text{N}$  сказывается в возникновении более легких изотопов C, Li, Be и B, а также d и n при взаимодействии космических лучей в атмосфере.

Кластерная структура легких ядер является ключевым аспектом в широком разнообразии реакций и может оказаться источником для поисков разреженных ядерных состояний. Она определяет процессы ядерной астрофизики и ядерные аспекты физики космических лучей и субнуклонных степеней свободы в ядерной материи. Учет кластерного характера ядерной фрагментации важен при исследованиях по радиационной медицине и геологии.

В настоящее время исследований по ядерной физике на ускорителях стали одним из наиболее быстро развивающихся разделов физики микромира. Детальное исследование явления фрагментации быстро движущихся ядер-снарядов имеет важные преимущества по сравнению с классическими экспериментами по расщеплению ядер-мишеней. Исследование кластеризации в релятивистском подходе может дать особые преимущества, поскольку в конечных состояниях фрагментов в



условиях очень малых передач энергии–импульса должна наиболее полно отражаться структура начальных состояний ядер.

Современные эксперименты проводятся в пучках релятивистских радиоактивных ядер на магнитных спектрометрах характерных для физики высоких энергий. Они ориентированы на регистрацию фрагментов с начальным зарядом как у исследуемого ядра или близким к нему. Однако в таком подходе теряются принципиально важные каналы, содержащие фрагменты He и H, в том числе из распадов  $^8\text{Be}$  и  $^9\text{Be}$ . Возможность разрешения этой проблемы методом ядерной эмульсии мотивировала эксперимент БЕККЕРЕЛЬ.

Поиск взаимодействий и измерение следов в облученных слоях ЯЭ требует высококвалифицированного и кропотливого труда с применением прецизионных оптических микроскопов. Выбор этого метода становится оправданным благодаря уникальной полноте наблюдения следов заряженных частиц с разрешением не менее 0.5 мкм. В слоях ядерной эмульсии, продольно облученных исследуемыми ядрами, направления следов определяются с исчерпывающей полнотой при наилучшем разрешении. Возможен одновременный анализ релятивистских фрагментов и сопровождающих фрагментов мишени практически без порогов детектирования.

Результаты данной работы могут использоваться для проверки теоретических моделей и служить отправным пунктом для исследования изотопов с большим избытком нейтронов. Полученный опыт указывает на перспективность применения метода ядерной эмульсии на основе моторизованного микроскопа с релятивистскими ядрами, а также ядерной физике низких энергий и радиационной дозиметрии. Эти возможности ярко раскрываются в тех поисковых исследованиях, требования которых выходят за возможности электронных детекторов.

**Целью работы** является сравнительный анализ данных эксперимента БЕККЕРЕЛЬ на нуклотроне ОИЯИ по релятивистской фрагментации ядер  $^{10}\text{Be}$  и  $^9\text{Be}$  в ядерной эмульсии для выявления особенностей фрагментации ядер мишени из состава ядерной фотоэмульсии.

**Задачи диссертационной работы:**

- для  $\alpha$ -частичных каналов фрагментации исследуемых ядер представить данные эксперимента БЕККЕРЕЛЬ по сопровождающим фрагментам ядер мишени;
- на основе этих данных исследовать возможную универсальность фрагментации ядер мишени под действием периферических нуклонов в исследуемых ядрах;
- провести подобный анализ в отношении каналов диссоциации с идентифицированными распадами ядер  $^8\text{Be}$  и  $^9\text{Be}$ ;
- дополнить статистику  $2\alpha$ -частичной диссоциации ядра  $^9\text{Be}$  и выполнить для них измерения углов испускания  $\alpha$ -частиц с использованием просмотровых и измерительных микроскопов МБИ-9 и KSM-1;
- представить результаты исследования в программном пакете ROOT.



# 1 Применение метода ядерной эмульсии в физике релятивистских ядер

## 1.1 Развитие исследований фрагментации релятивистских ядер

Основы метода ЯЭ в применении к релятивистским частицам были разработаны в тот начальный период, когда основополагающие открытия в физике высоких энергий делались при исследовании космических лучей. Эти достижения представлены в книге «Исследование элементарных частиц фотографическим методом» С. Пауэлла, П. Фаулера, Д. Перкинса [1]. Ее завершающая глава посвящена изучению следов релятивистских ядер в слоях ЯЭ, облучавшихся в стратосфере, и порожденных ими ядерных звезд. Некоторые из наблюдаемых событий ядерных взаимодействий, содержащие по несколько следов релятивистских  $\alpha$ -частиц, явным образом указывали на кластеризацию.

В 70-х гг. начались облучения стопок ЯЭ легкими ядрами с энергией несколько ГэВ на нуклон, ускоренными на Синхрофазотроне ОИЯИ и Бевалаке (Беркли, США), а в 90-х гг. средними и тяжелыми на AGS (BNL) и SPS (CERN) при существенно более высоких значениях энергии. Метод ЯЭ с разрешением 0.5 мкм обеспечил эксклюзивные наблюдения треков от тяжелых релятивистских ядер и вплоть до однозарядных частиц и медленных осколков ядер мишени. Поиск звезд по первичным следам позволил сделать обзор всех возможных типов взаимодействия без выборки и установить общие закономерности реакций.

Особенности ядер проявились в конусе релятивистской фрагментации. К ним можно отнести универсальное образование пар  $\alpha$ -частиц с предельно малыми углами разлета - порядка несколько мрад. Они указали на возможность изучения кластеризации, начиная с распадов  ${}^8\text{Be}$ . В принципе коллимация релятивистских фрагментов, рожденных в периферическом столкновении, позволяет детектировать их все сразу в малом телесном угле. Однако электронные эксперименты натолкнулись на сложности, обусловленные квадратичной зависимостью ионизации от зарядов, крайне малой угловой расходимостью релятивистских фрагментов, и, зачастую, примерным совпадением по магнитной жесткости с ядрами пучка. До сих пор полное детектирование ансамблей легчайших релятивистских фрагментов демонстрирует только метод ЯЭ.

В 2002 г. вновь образованным сотрудничеством БЕККЕРЕЛЬ возобновлена программа облучения на ускорительном комплексе Нуклотрон ОИЯИ стопок ЯЭ в пучках релятивистских изотопов бериллия, бора, углерода и азота, в том числе радиоактивных (рисунок 1.1). Совпадая с именем ученого, название проекта БЕККЕРЕЛЬ указывает на задачи – *Beryllium (Boron) Clustering Quest in Relativistic Multifragmentation* [2]. На данных этих облучений основана и настоящая работа.

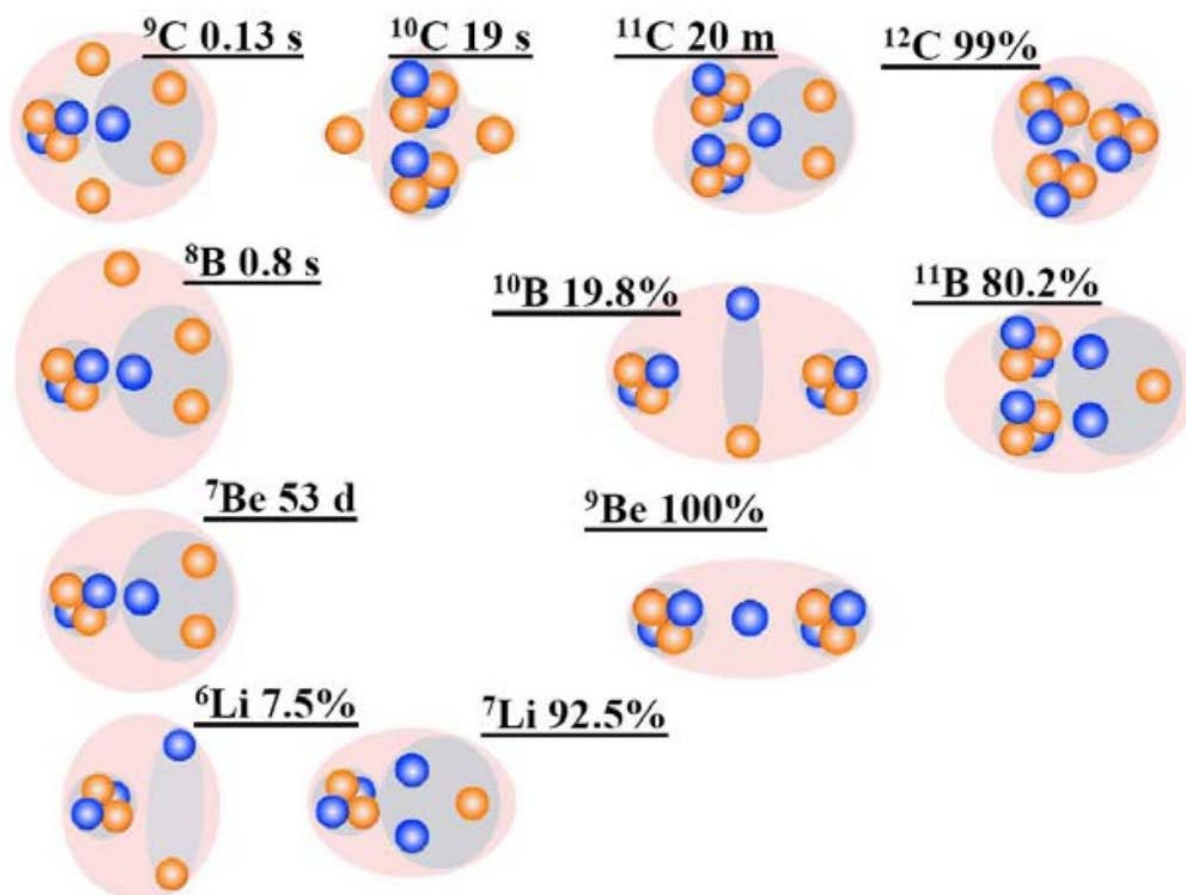


Рисунок 1.1 – Диаграмма кластерных степеней свободы в стабильных и нейтронодефицитных ядрах. Указаны распространенности изотопов или времена жизни; красные кружки соответствуют протонам, синие - нейтронам; серым фоном выделены кластеры

Создание пучка  $^{10}\text{B}$ , является неординарной задачей. Ядра  $^{10}\text{B}$  создаются под воздействием мощного импульса  $\text{CO}_2$  - лазера. Вещество лазерной мишени должно обладать высокой термической стойкостью и не содержать ядер  $^{12}\text{C}$  и  $^{16}\text{O}$ , имеющих те же отношения заряда к весу. Поэтому в качестве лазерной мишени был использован гексаборид лантана  $\text{LaB}_6$  – тугоплавкое вещество, используемое в катодах электронных микроскопов. В диссоциации  $^{10}\text{B}$  была установлена кластеризация  $2\alpha d$ . Ядро  $^9\text{Be}$ , обладающее свойствами слабосвязанной системы  $2\alpha n$ , является «краеугольным камнем» кластерной физики и может служить наиболее удобным источником нестабильных ядер  $^8\text{Be}$ . Однако токсичность бериллия делает невозможным непосредственное ускорение ядер  $^9\text{Be}$ . Поэтому  $^9\text{Be}$  был получен путем настройки магнитооптического канала транспортировки выведенного пучка нуклотрона на сепарацию продуктов фрагментации  $^{10}\text{B} \rightarrow ^9\text{Be}$ .

## 1.2 Возможности метода

Поиск в ЯЭ ядерных взаимодействий без выборки (или метод «по следу»), обеспечивает достаточно равномерную эффективность обнаружения всех возможных типов. Будучи достаточно трудоемким, он обеспечивает наилучшее качество и систематичность просмотра и применяется при исследовании, проходящем впервые. Статистика в несколько десятков и даже

сотен периферических взаимодействий с определенными конфигурациями релятивистских фрагментов достижима при поперечном сканировании.

Релятивистские фрагменты сосредоточены в конусе (1)

$$\sin\theta_{fr} = p_{fr}/P_0, \quad (1)$$

где  $p_{fr} = 0.2$  ГэВ/с – величина, характеризующая Ферми-импульс нуклонов в ядре-снаряде,

$P_0$  – его импульс на нуклон.

Заряды релятивистских фрагментов  $Z_{fr} = 1$  и  $2$ , наиболее важных в проблеме нестабильных состояний, определяются визуально благодаря явной разнице в ионизации. Треки фрагментов с зарядом  $Z_{fr} \geq 3$  идентифицируются по плотности проявленных зерен вдоль следа или по плотности образовавшихся  $\delta$ -электронов. Условием отбора периферических взаимодействий, составляющих несколько процентов от неупругих взаимодействий, является примерное сохранение релятивистскими фрагментами заряда ядер пучка  $Z_{pr}$ , т. е.  $Z_{pr} = \sum Z_{fr}$ . Такое условие применяется в облучениях ЯЭ во вторичных пучках, представляющих собой смесь двух или трех изотопов.

При измерительной базе 1 мм разрешение для следов релятивистских фрагментов не хуже  $10^{-3}$  рад. Поперечный импульс  $P_T$  фрагмента с массовым числом  $A_{fr}$  определяется как  $P_T \approx A_{fr}P_0\sin\theta$  в приближении сохранения им скорости первичного ядра (или  $P_0$ ). Во фрагментации ядер, составляющих ЯЭ, могут наблюдаться  $b$ -частицы ( $\alpha$ -частицы и протоны с энергией ниже 26 МэВ),  $g$ -частицы (протоны с энергией свыше 26 МэВ), а также  $s$ -частицы (рожденные мезоны). Наиболее периферические взаимодействия, именуемые когерентной диссоциацией или “белыми” звездами, не сопровождаются фрагментацией ядер мишени и рождением мезонов ( $s$ -частиц). Доступны фотографии и видеозаписи характерных взаимодействий.

Присвоение массовых чисел трекам фрагментов Н и Не возможно по суммарным значениям импульса, полученным из среднего угла кулоновского рассеяния. Использование этого трудоемкого метода оправдано в особых случаях для ограниченного числа следов, так как для этого минимальная длина трека должна составлять несколько сантиметров. В случае диссоциации стабильных ядер достаточно предположить соответствие Не –  $^4\text{He}$  и Н –  $^1\text{H}$ , поскольку установленные вклады  $^3\text{He}$  и  $^2\text{H}$  не превышают 10%. Это упрощение особенно верно в случае распадов  $^8\text{Be}$  и  $^9\text{Be}$  в предельно узкие струи  $\alpha$ -частиц и протонов [3].

Инвариантная масса системы релятивистских фрагментов определяется как сумма всех скалярных произведений 4-импульсов  $P_{i,k}$  фрагментов. Вычитание массы начального ядра или суммы масс фрагментов  $Q = M^* - M$  является вопросом удобства представления результатов. Компоненты  $P_{i,k}$  определяются в приближении сохранения фрагментами начального импульса на нуклон.

Согласно ширинам,  $^8\text{Be}$  и  $^9\text{Be}$  могут являться полноценными участниками в релятивистской фрагментации. Они распадаются на нескольких тысячах ( $^8\text{Be}$ ) или нескольких десятках ( $^9\text{Be}$ ) атомных размеров, т.

е. за время на много порядков большее, чем время возникновения других фрагментов. Вследствие наименьшей энергии распада  ${}^8\text{Be}$  и  ${}^9\text{B}$  должны проявляться как пары и тройки следов  $\text{He}$  и  $\text{H}$  с наименьшими углами раскрытия, что выделяет их на фоне других даже на этапе поиска.

Реконструкция по инвариантной массе распадов  ${}^8\text{Be}$  позволяет сравнивать данные при отличающихся значениях первичной энергии ускоренных ядер. Энергия распада  ${}^8\text{Be}$  из основного состояния  $0^+$  на  $2\alpha$ -частицы составляет 91.8 кэВ, а ширина  $5.57 \pm 0.25$  эВ [4]. В диссоциации  ${}^9\text{Be} \rightarrow 2\alpha$  при 1.2 А ГэВ уточнен отбор событий с образованием ядра в  ${}^8\text{Be}$  в основном состоянии по величине  $Q_{2\alpha}$ . (обзор [3]). Распределение по  $Q_{2\alpha}$  для измеренных 500  $2\alpha$ -пар, в том числе 198 “белых”, указывает на ограничение  $Q_{2\alpha}({}^8\text{Be}) \leq 0.2$  МэВ. Оно включает принятое приближение сохранения вторичными фрагментами импульса родительского ядра, кинематический эллипс распада  ${}^8\text{Be}$  и разрешение угловых измерений. Ядро  ${}^8\text{Be}$  является неизменным продуктом распада  ${}^9\text{B}$ . Основное состояние ядра  ${}^9\text{B}$  выше порога  ${}^8\text{Be}p$  на 185.1 кэВ, а его ширина  $0.54 \pm 0.21$  кэВ [4]. При изучении когерентной диссоциации изотопа  ${}^{10}\text{C}$  при энергии 1.2 А ГэВ канал  $2\text{He}2\text{H}$  составил 82% “белых” звезд, среди которых условию  $Q_{2\alpha p}({}^9\text{B}) < 0.5$  МэВ отвечает  $(30 \pm 4)\%$ . Во всех этих  $2\alpha p$ -тройках установлена последовательность распадов  ${}^9\text{B} \rightarrow {}^8\text{Be} + p$  и  ${}^8\text{Be} \rightarrow 2\alpha$ .

### 1.3 Особенности релятивистской фрагментации ${}^6\text{Li}$ , ${}^{10}\text{B}$ , ${}^{14}\text{N}$

Ядро  ${}^6\text{Li}$  является единственным, не считая дейтрона, стабильным ядром с экзотической структурой. Из-за увеличенных размеров и слабой связи нуклонов экзотические ядра характеризуются усиленными сечениями взаимодействия и суженными импульсными распределениями их основ во фрагментации. Эти свойства проявились во фрагментации  ${}^6\text{Li}$ , детально изучавшееся в ФИАН (Физический институт им. П.Н. Лебедева Российской академии наук). Была исследована сопровождающая множественность. Методом многократного рассеяния выполнена идентификация изотопов  $\text{He}$  и  $\text{H}$ .

Средний пробег ядер  ${}^6\text{Li}$  до неупругого взаимодействия оказался примерно на 3 см меньше, чем следует из формулы геометрического перекрытия ( $\approx 17$  см). Он указывает на большой радиус ядра  ${}^6\text{Li}$ , равный  $(2.7 \pm 0.1)$  Ферми. Будучи аномально большим, это значение при нормальной плотности отвечает ядру с  $A = 11$ . Особенностью изотопного состава фрагментов  ${}^6\text{Li}$  стал необычно большой выход дейтронов, практически равный выходу протонов. Для фрагментации  ${}^6\text{Li} \rightarrow \alpha$  значение среднего поперечного импульса  $\alpha$ -частиц оказалось пониженным –  $\langle p_T^\alpha \rangle = (0.13 \pm 0.01)$  ГэВ/с, в то время как в диссоциации ядер  ${}^{12}\text{C} \langle p_T^\alpha \rangle = (0.24 \pm 0.01)$  ГэВ/с. Статистика фрагментов  $\alpha/ap/ad$ , составившая 97/65/61, не указывает на выделенный вклад нейтронов. Как «золотые» события можно рассматривать 31 «белую» звезду  $ad$ . Среди них 23 события соответствуют диссоциации ( $E_{\text{th}} = 1.47$  МэВ), по 4 события в каналах  ${}^3\text{He}t$  ( $E_{\text{th}} = 15.8$  МэВ) и  $tdp$  ( $E_{\text{th}} = 21.2$

МэВ). По-видимому, структура  ${}^6\text{Li}$  – это пара  $\alpha d$ , что, в частности, определяет захват тепловых нейтронов дейтронным кластером [5].

Для  ${}^{10}\text{B}$  пороги отделения нуклонов и легчайших ядер близки –  $E_{\text{th}}({}^6\text{Li}\alpha) = 4.5$  МэВ,  $E_{\text{th}}({}^8\text{Be}d) = 6.0$  МэВ,  $E_{\text{th}}({}^9\text{Be}p) = 6.6$  МэВ. В эксперименте БЕККЕРЕЛЬ установлено, что «белые» звезды  $2\text{He}+H$   ${}^{10}\text{B}$  при энергии 1 ГэВ на нуклон составляют 65%. Канал  $\text{He}+3\text{H}$  составляют 15%.  $\text{Li}+\text{He}$  – 10%, а  ${}^9\text{Be}+p$  – всего 2%. Спин  ${}^{10}\text{B}$  равен 3. Удаление протона с  $p$ -оболочки ведет к образованию  ${}^9\text{Be}$  со спином 3/2 и не требует передачи углового момента. Тем не менее, канал  ${}^9\text{Be}+p$  подавлен, что указывает на основу  $2\alpha$ . Наличие фрагментов ядра-мишени практически не сказывается на фрагментации при сохранении заряда 5. В целом доминируют каналы, содержащие только  $\text{He}$  и  $\text{H}$  (90%), при лидировании  $2\text{He}+H$  (76%). В 24% событий в канале  $2\text{He}+H$  содержат распады  ${}^8\text{Be}$ , а 12% – распады  ${}^9\text{B} \rightarrow {}^8\text{Be}+p$ . В «белых» звездах  ${}^{10}\text{B} \rightarrow 2\text{He}+H$  методом многократного рассеяния с соотношением  $2 \pm 0.25$  были частично идентифицированы  $p$  и  $d$  [6].

Изучение  ${}^{14}\text{N}$  – промежуточного ядра между кластерным ядром  ${}^{12}\text{C}$  и дважды магическим  ${}^{16}\text{O}$  – указывает на усложнение ядерной структуры. В событиях с  $\Sigma Z_{\text{fr}} = 7$  основной вклад обеспечивают каналы  $\text{C}+H$ ,  $3\text{He}+H$  и  $2\text{He}+3\text{H}$  (77%). Доля  $\text{C}+H$  ( $E_{\text{th}} = 7.6$  МэВ) – 25%, а  $\text{B}+\text{He}$  ( $E_{\text{th}} = 20.7$  МэВ) – 8%. Заметно снизилась доля  $d$  по отношению к  $p$  протонам по сравнению с  ${}^6\text{Li}$  и  ${}^{10}\text{B}$ . Среди «белых» звезд и событий с фрагментами мишени лидируют  $3\text{He}+H$  ( $E_{\text{th}} = 15$  МэВ) – 35%. Тем самым  ${}^{14}\text{N}$  – источник ансамблей  $3\alpha$ , соответствующих возбуждениям  ${}^{12}\text{C}$  от порога 7.3 до 14 МэВ.  ${}^{14}\text{N}$  с вероятностью не менее 20% фрагментирует в  $3\text{He}+H$  через  ${}^8\text{Be}$  [7].

Статистика  $3\text{He}+H$  лидирует как среди «белых» звезд  $N_{\text{ws}}(3\text{He}+H)$ , так и среди взаимодействий с фрагментами мишени  $N_{\text{tf}}(3\text{He}+H)$ . В «белых» звездах  $\text{C}+H$  и  $3\text{He}+H$  методом многократного рассеяния с отношением  $1.9 \pm 0.7$  были частично идентифицированы  $p$  и  $d$ . Несмотря на большие статистические ошибки это значение позволяет предположить, что роль дейтрона как кластера, так же как и для  ${}^{10}\text{B}$ , снижается по сравнению со случаем  ${}^6\text{Li}$ . Для событий с фрагментами  $3\text{He}+H$  и  $3\text{He}$ , которые выглядят как реакция срыва нейтрона или протона с разрушением мишени, имеется существенное различие чисел  $N_{\text{tf}}$ . Их отношение равно  $N_{\text{tf}}(3\text{He}+H)/N_{\text{tf}}(3\text{He}) = 1.9 \pm 0.2$ , а в схожем канале  ${}^{10}\text{B} \rightarrow 2\text{He}+H$  –  $N_{\text{tf}}(2\text{He}+H)/N_{\text{tf}}(2\text{He}) = 2.1 \pm 0.3$  [8]. Возможность использования этих отношений для прояснения структур  $2\alpha$  и  $3\alpha(+pn)$  послужила мотивацией для анализа фрагментации ядер мишени в периферических взаимодействиях этих ядер. Данные по фрагментации  ${}^9\text{Be} \rightarrow 2\text{He}+n$  с фрагментами мишени  $N_{\text{tf}}$  могут служить отправными.

#### 1.4 Метод ядерных фотоэмульсий

Метод ядерных фотографических эмульсий используется для регистрации и анализа следов заряженных частиц практически любой энергии. По измерениям характеристик этих следов можно сделать выводы: о числе и природе образующихся при расщеплении ядра заряженных частиц, об их угловых и энергетических распределениях, о энергии и импульсе

передаваемых ядру при столкновении и т.д. Метод ядерных эмульсий заключается в том, что при прохождении через фотоэмульсию заряженная частица активирует на своем пути кристаллы галоидного серебра и делает их способными к проявлению (латентное изображение). После специальной химической обработки эмульсионных слоев в них появляются следы частиц в виде цепочки проявленных зерен (трек), хорошо различимых под оптическим микроскопом.

К достоинствам методики можно отнести то, что ядерная эмульсия регистрирует все заряженные частицы, проходящие через нее в течение определенного промежутка времени, так как время чувствительности фотоэмульсии, по сути, определяется временем экспозиции, которое может быть от нескольких секунд до нескольких недель. По точности измерения малых углов метод ядерных эмульсий не имеет себе равных. Этот метод обладает рекордным пространственным разрешением, который с успехом применяется для определения углов разлета частиц, фрагментов и ядер, образующихся в результате ядерного взаимодействия. Еще одним достоинством метода ядерной фотографической эмульсии является то, что при проведении экспериментов на ускорителях он не требует больших затрат времени работы самого ускорителя. Расходы на аппаратуру также сравнительно невелики. Положительной чертой исследования процесса фрагментации ядер с помощью эмульсии является то, что следы частиц в желатине остаются навсегда и исследователь может изучать их снова и снова, чего не невозможно сделать машинным методом. Объем накопленной информации в слое ядерной эмульсии площадью  $125 \text{ см}^2$  оценивается на уровне 10 ТБ. Для сравнения, dvd диск площадью  $111 \text{ см}^2$  вмещает в себя 9 Гб информации.

Среди множества достоинств данного метода, имеется ряд недостатков. А именно, сложность состава вещества фотоэмульсии не позволяет точно определить с каким ядром произошло взаимодействие налетающей частицы. Его устраняют введением дополнительных элементов и критериев отбора обрабатываемых событий. Одним из главных недостатков метода является невысокая скорость обработки слоев ядерной эмульсии и как следствие необходимость длительного времени для набора статистики событий. Как правило, это компенсируется получаемым результатом. Тем самым, позволяя проводить планирование электронных экспериментов.

Тем не менее стоит отметить, что в последнее время ведутся разработки полностью автоматизированных считывающих комплексов, с целью ускорения процедуры сканирования слоев ядерных эмульсий. Например, в Нагойском университете разработана система Hyper Track Selector (HTS), способная сканировать слои ядерных эмульсий со скоростью  $4000 \text{ см}^2/\text{ч}$  [9], в то время как ручной метод сканирования ограничен скоростью в несколько  $\text{мм}^2$  в час.

Во время исследований следует точно знать состав эмульсии в реальных условиях ее облучения, так как это может существенно влиять на точность измерений, которая может быть достигнута при помощи

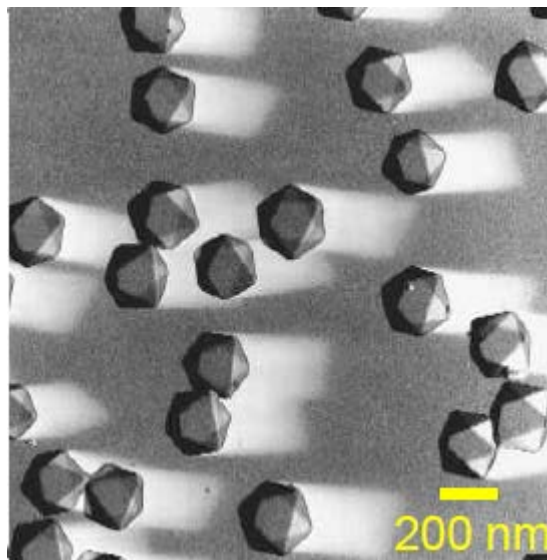


фотометода и которая требуется при точном определении энергии однородной группы частиц по их средним пробегаем. Одним из препятствий, возникающих при определении химического состава эмульсий, является то, что желатина достаточно быстро обменивается водой с находящейся с ней в контакте газообразной или жидкой средой. Это затрудняет определение плотности при помощи обычного метода, основанного на взвешивании образца сначала в воздухе, а затем в жидкости с известной плотностью.

Тем не менее значительную роль в развитии существующих представлений о механизме взаимодействия частиц большой энергии с ядрами сыграло использование ядерных фотографических эмульсий.

### **1.5 Общая характеристика ядерной фотоэмульсии**

Ядерные эмульсии представляют собой высококонцентрированные и высокодисперсные эмульсии из бромистого серебра. Концентрация  $\text{AgBr}$ , который находится в форме микрокристаллов почти сферической формы пример представлен на рисунке 1.2, в большинстве изготавливаемых в настоящее время эмульсий составляет около 83% по весу. В качестве среды, в которой распределены микрокристаллы  $\text{AgBr}$ , наиболее широко используется желатина с пластификатором. Средние размеры микрокристаллов бромистого серебра для большинства ядерных эмульсий находятся в пределах 0.12-0.3 мкм, эмульсии НИКФИ, Ильфорд, Кодак, Агфа, Истмен и др. Однако, в последнее время в Японии начало развивается производство наноразмерных ЯЭ, размер кристаллов в которых составляет несколько десятков нм. Такие эмульсии предполагается использовать в экспериментах по прямой регистрации частиц темной материи (WIMP), длина трека ядра отдачи при взаимодействии с WIMP в ЯЭ оценивается в 100-200 нм.



**Рисунок 1.2 – Микрофотография кристаллов галогенида серебра под микроскопом.  
Типичный размер кристалла галогенида серебра в стандартной фотоэмульсии  
составляет 0.2 мкм**

Удельная плотность всех эмульсий с нормальным составом находится в пределах  $3.8 - 4.0 \text{ г/см}^3$ . По причине гигроскопичности желатины удельный вес, состав и усадка ЯЭ сильно зависит от влажности. Для постоянства перечисленных параметров ЯЭ необходимо строго соблюдать микроклиматический режим производственного помещения при относительной влажности 60%.

Способ производства ЯЭ зависит от конкретной задачи, для которой ЯЭ будут применяться. Выделяют следующие типы ЯЭ. Бесподложечные ЯЭ с толщиной эмульсионного слоя 400-600 мкм. Производство бесподложечных ЯЭ представляет собой достаточно сложную технологическую задачу, однако оно оправдано применением таких ЯЭ в экспериментах, где необходимо использовать большой чувствительный объем. Второй тип ЯЭ - слои, толщиной эмульсионного слоя от 50 до 400 мкм, нанесенные на подложку (пленка или стекло).

Анализ элементарного состава ядерных эмульсий показывает, что в эмульсии главным образом содержатся три отстоящих по атомному весу группы атомов: Н - С, N, О - Ag, Вг при весьма незначительном количестве атомов S, J.

Основными компонентами ядерной эмульсии являются:

а) – галоидное серебро – главным образом бромистое, обладающее плотностью  $6.47 \text{ г/см}^3$ . Галоидное серебро находится в эмульсии в виде микрокристаллов (или «зерен») кубической или почти сферической формы в зависимости от метода приготовления.

б) – желатин и пластификатор (например, глицерин). Желатин и пластификатор содержат следующие элементы: углерод, азот, кислород, водород и серу. Желатин представляет собой сложное вещество, вследствие чего неизбежно существует незначительная разница в химическом составе желатина, поставляемой различными изготовителями. Глицерин вводится в эмульсию в качестве пластифицирующего вещества. Он уменьшает хрупкость эмульсии и способствует появлению пластических деформаций при возникновении напряжений в слое. Последнее имеет особенно большое значение в тех случаях, когда пластинки должны экспонироваться в вакууме, поскольку в подобных условиях эмульсия без пластификатора часто отрывается от стекла. Сцепление между эмульсией и стеклом настолько велико, что последнее часто разрушается под действием натяжений, появляющихся при высыхании эмульсии, причем из пластинок вырываются чешуйки стекла, прочно прилипшие к эмульсии.

в) – вода.

По своим фотографическим характеристикам все ядерные эмульсии разделяются на три основных типа [10].

а) – Эмульсии низкой чувствительности, предназначенные для регистрации сильноионизирующих частиц с потерями энергии  $\geq 50 \text{ кэВ}$  на 1мкм пробега (осколков деления, многозарядных ионов, медленных  $\alpha$ -частиц).

б) – Эмульсии средней чувствительности, способные регистрировать следы частиц с ионизацией  $\geq 2\text{кэВ}$  на 1мкм пробега (протоны с энергией 50-100МэВ).

в) – Эмульсии высокой чувствительности, регистрирующие следы частиц с минимальной ионизацией (0,55 кэВ на 1 мкм).

Эмульсии, применяемые в настоящее время для регистрации следов заряженных частиц, отличаются от обычных фотоэмульсий двумя особенностями:

а) – отношение количества галоидного серебра к желатине в ядерной эмульсии приблизительно в восемь раз больше, чем в обычной;

б) – толщина слоя ядерной эмульсии превышает, как правило, в 10 – 100 раз толщину обыкновенной эмульсии. Кроме того, в современных условиях ядерные эмульсии часто применяются в виде отдельных слоев, лишенных стеклянной подложки.

Одной из основных используемых ядерных фотоэмульсий является эмульсия типа БР-2, обеспечивающая чувствительность вплоть до релятивистских однозарядных частиц и дающих минимум ионизационных потерь. Основные характеристики этой эмульсии приведены в таблице 1.

Таблица 1 – Химический состав эмульсии (литературные данные по составу ядерной фотоэмульсии БР-2; относительная влажность 58%) [1]

Элемент	Z	$10^{22}$ атомов/см <sup>3</sup>
Серебро/ Ag	47	1.02
Бром/ Br	35	1.01
Йод/ I	53	0.006
Углерод/ C	6	1.39
Водород/ H	1	3.19
Кислород/ O	8	0.94
Сера/ S	16	0.014
Азот/ N	14	0.32

Выбор типа эмульсии зависит от ее предполагаемого применения. Чувствительность эмульсии должна быть достаточно велика, чтобы обеспечить плотность зерен, необходимую для регистрации следа частицы. При излишней чувствительности плотность зерен может оказаться настолько большой, что отдельные зерна станут неразличимыми и их нельзя будет сосчитать.

## 1.6 Механизм образования следов

В процессе образования следов заряженных частиц в эмульсии решающую роль играет возникновение  $\delta$ -электронов (электроны, выбитые из атомов в процессе ионизации и в свою очередь способные ионизировать атомы среды, через которую они движутся) пример на рисунке 1.3, при наличии которых в данном зерне может выделяться количество энергии, намного превышающее то максимальное значение, которое оказалось бы возможным, если бы энергетические потери частиц были равномерно распределены вдоль ее пути. Так, например, в минимуме ионизации удельная ионизация частицы с зарядом составляет около 700 кэВ/мм. При равномерных потерях максимальное количество энергии, которое может быть поглощено зернами диаметром 0,3  $\mu$ , должно быть порядка 300 эВ (при прохождении частицы по диаметру зерна). С другой стороны, пробег  $\delta$ -электрона с энергией меньше 5 кэВ оказывается настолько коротким, а его траектория настолько искривленной вследствие рассеяния, что при возникновении такого электрона внутри зерна он может остановиться, не выходя за пределы последнего. Таким образом, в отдельных зернах выделяется значительно большее количество энергии, которое может быть затрачено на образование скрытого изображения. Вследствие описанного эффекта оказывается возможным, что почти во всех эмульсиях некоторые зерна станут способными к проявлению после прохождения через них одной частицы, удельная ионизация которой близка к минимальной; однако для образования следа, достаточно хорошо различимого под микроскопом, число таких зерен на единицу длины траектории должно быть довольно велико по сравнению с общим «фоном». Изображения событий множественной фрагментации релятивистских ядер были получены еще в 40х годах в слоях ядерной эмульсии при воздействии космических лучей в стратосфере [11]. С коллекцией фотографий ядерных взаимодействий в ядерной эмульсии и их полным описанием можно ознакомиться в фундаментальном труде С. Пауэлла, П. Фаулера и Д. Перкинса [1].

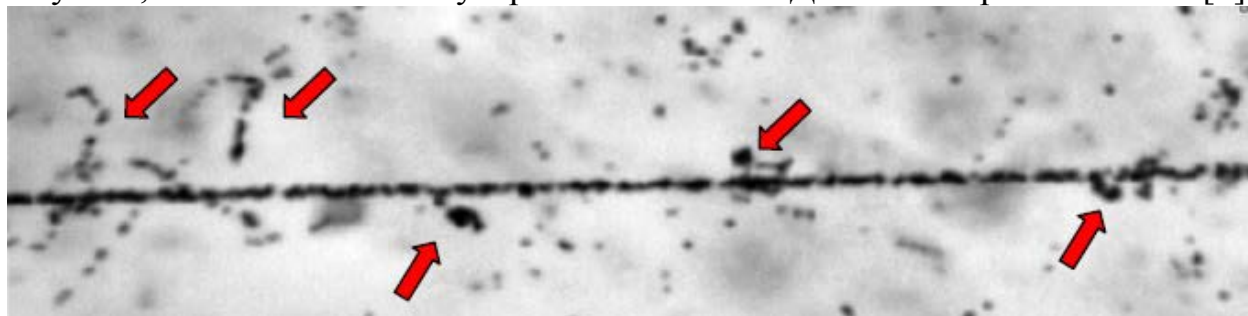


Рисунок 1.3 – Макрофотография трека релятивистского ядра  $^{14}\text{N}$  в ядерной эмульсии. Вдоль трека наблюдаются следы образованных  $\delta$ -электронов, указанные стрелками

При анализе данных в фотоэмульсионной методике заряженные частицы принято делить на 4 типа, в зависимости от их относительной ионизации  $I/I_0$  и скорости  $\beta$ :

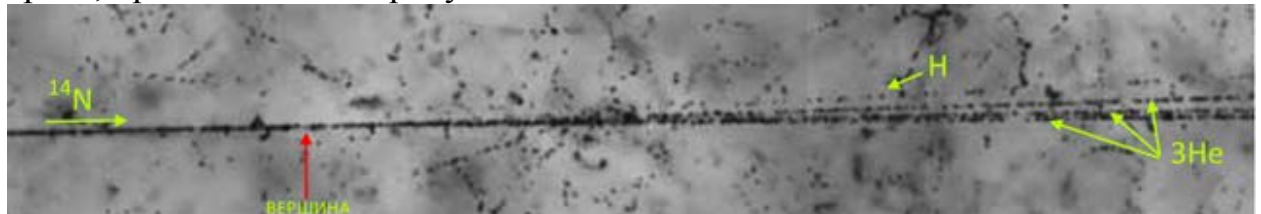
а) – “черные” ( $b$ -частицы) - представляют собой следы фрагментов ядра мишени с относительной ионизацией  $I/I_0 \geq 7.0$  и  $\beta < 0.23$ , где  $I_0$  - ионизация

на треках релятивистских частиц с зарядом  $Z = 1$ . Однако в практическом плане,  $b$ -частицы зачастую удобно идентифицировать по величине их пробега в объеме ядерной фотоэмульсии -  $L \leq 0.3$  см;

б) – “серые” ( $g$  - частицы) - в основном это протоны, выбитые из ядра-мишени, с относительной ионизацией  $6.8 > I/I_0 \geq 1.4$  и  $\beta < 0.7$ , с остаточным пробегом  $\geq 0.3$  мм. Также к этому типу частиц относится небольшая примесь  $\pi$ -мезонов, зависящая от начальной энергии взаимодействия;

в) – “релятивистские” (ливневые или  $s$  - частицы) - к этому классу заряженных частиц относят провзаимодействовавшие протоны налетающего ядра и однозарядные ( $Z=1$ ) непровзаимодействовавшие фрагменты налетающего ядра с относительной ионизацией  $I/I_0 < 1.4$  и  $\beta > 0.7$ .

г) – “фрагменты” ( $f$  - частицы) - многозарядные фрагменты налетающего ядра с зарядом  $Z \geq 2$ . Они не входят в число  $b$ - и  $g$  - частиц которым соответствует по производимой ими ионизации. Треки релятивистских однозарядных частиц и фрагменты ядра-снаряда с  $Z=2$  легко различимы под микроскопом числом проявленных зерен на единицу длины трека, представлены на рисунке 1.4.



**Рисунок 1.4 – Фотография события фрагментации релятивистского ядра  $^{14}\text{N}$  с образованием в узком переднем конусе трех фрагментов He и одного H без наблюдения следов ядра-мишени в широком конусе**

## 2 Топология фрагментации налетающего ядра

### 2.1 Фрагментация ${}^9\text{Be} \rightarrow 2\text{He}$

Изотоп  ${}^9\text{Be}$  является объектом значительного интереса для исследований в области кластерных степеней свободы в физике легких ядер. Отличаясь от изотопа  ${}^8\text{Be}$  на один нейтрон, ядро  ${}^9\text{Be}$  представляет собой слабосвязанную систему  $2\alpha+n$ , что делает его особенно интересным для изучения. Энергетический порог развала  ${}^9\text{Be} \rightarrow 2\alpha+n$  (рисунок 2.1) составляет 1.57 МэВ. В оболочечной модели конфигурация основного состояния  ${}^9\text{Be}$  представлена как  $(1s1/2)^4(1p3/2)^5$ , а его спин и четность равны  $J^\pi = 3/2^-$  [12].

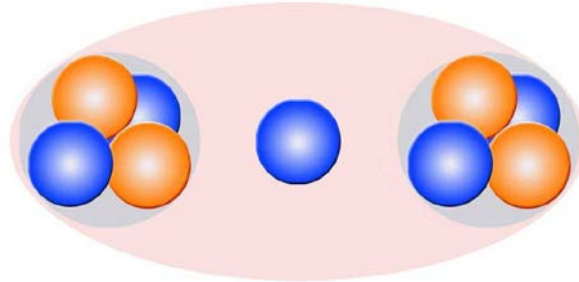
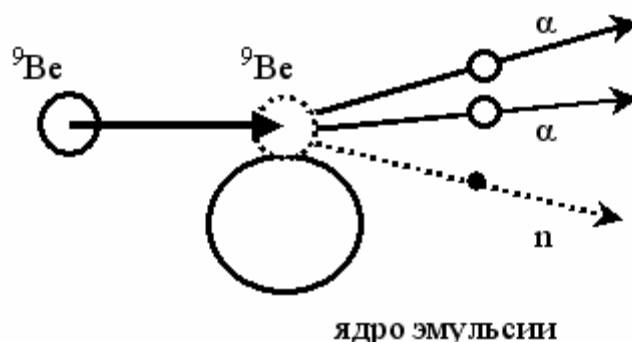


Рисунок 2.1 – Схематичное изображение кластерной структуры ядра  ${}^9\text{Be}$  ( $2\alpha+n$ )

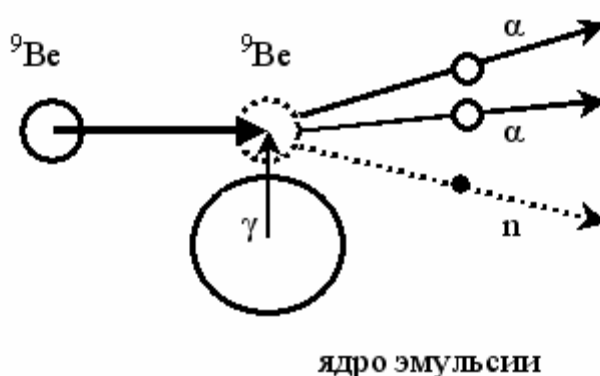
Благодаря низкому порогу отделения нейтрона диссоциация ядер  ${}^9\text{Be}$  может служить наиболее удобным источником нестабильных ядер  ${}^8\text{Be}$  (рисунок 2.2). Изотоп  ${}^8\text{Be}$  известен как единственное ядро, основное состояние которого характеризуется как  $\alpha$ -частичный Бозе-конденсат. Исследование фрагментации ядра  ${}^9\text{Be}$  на пару  $\alpha$ -частиц кажется очевидным отправным пунктом к более сложным  $N\alpha$ -системам. В работах приводится сравнение теоретических описаний свойств изотопа  ${}^9\text{Be}$  на основе использования двух частичной ( ${}^8\text{Be} + n$ ) и трех кластерной ( $\alpha + \alpha + n$ ) моделей. Опираясь на данные модельные описания, можно получить вклад различных состояний  $2\alpha+n$  системы в структуру  ${}^9\text{Be}$ , оценить среднее расстояние между  ${}^8\text{Be}$  и нейтроном, получить значения магнитного и электрического квадрупольного моментов.

## Периферическая фрагментация

### Ядерные взаимодействия



### Электромагнитные взаимодействия



**Рисунок 2.2 – Схематическое изображение исследуемых процессов фрагментации  ${}^9\text{Be} \rightarrow 2\alpha + n$  на ядрах фотоэмульсии. Вылет нейтрона изображается прерывистой линией со стрелкой, поскольку нейтрон не оставляет трека в эмульсии**

В двух частичной (core+n) модели ядро  ${}^9\text{Be}$  может быть представлено в виде системы нейтрона в состоянии  $3/2^-$  и кора  ${}^8\text{Be}$  в состояниях  $0^+$  (g.s.) и  $2^+$  с порогами отделения нейтрона 1.67 и 4.71 МэВ, соответственно. Отличительной особенностью модели является незначительный вклад  $4^+$  состояния  ${}^8\text{Be}$ .

Интересным для проверки, является предположение о возможности наблюдения  ${}^8\text{Be}$  в состояниях  $0^+$  и  $2^+$ , при «мягком удалении» нейтрона из  $2\alpha+n$  системы  ${}^9\text{Be}$ . Образование изотопа  ${}^8\text{Be}$  в состояниях  $0^+$  и  $2^+$  может происходить в периферических процессах при фрагментации релятивистского ядра  ${}^9\text{Be}$  в  $2\alpha+n$  систему, наблюдаемую в конечном состоянии. Наряду с образованием не связанного  ${}^8\text{Be}$  существует возможность образования промежуточного ядра  ${}^5\text{He}$ . Соответствующий энергетический порог равен 2.44 МэВ. Продуктом диссоциации  ${}^5\text{He}+\alpha$  в конечном состоянии также будет являться  $2\alpha+n$  система.

Согласно ширинам,  ${}^8\text{Be}$  и  ${}^9\text{Be}$  могут являться полноценными участниками в релятивистской фрагментации. Они распадаются на нескольких тысячах ( ${}^8\text{Be}$ ) или нескольких десятках ( ${}^9\text{Be}$ ) атомных размеров, т.

е. за время на много порядков большее, чем время возникновения других фрагментов. Вследствие наименьшей энергии распада  ${}^8\text{Be}$  и  ${}^9\text{B}$  должны проявляться как пары и тройки следов He и H с наименьшими углами раскрытия, что выделяет их на фоне других даже на этапе поиска.

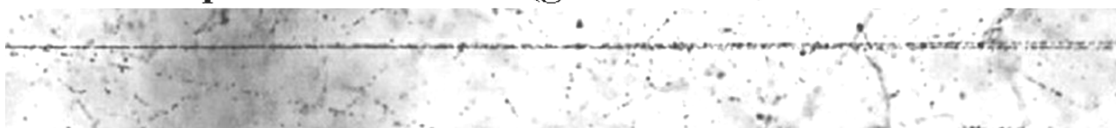
Реконструкция по инвариантной массе распадов  ${}^8\text{Be}$  позволяет сравнивать данные при отличающихся значениях первичной энергии ускоренных ядер. Энергия распада  ${}^8\text{Be}$  из основного состояния  $0+$  на  $2\alpha$ -частицы составляет 91.8 кэВ, а ширина  $5.57 \pm 0.25$  эВ [4]. В диссоциации  ${}^9\text{Be} \rightarrow 2\alpha$  при 1.2 А ГэВ уточнен отбор событий с образованием ядра в  ${}^8\text{Be}$  в основном состоянии по величине  $Q_{2\alpha}$ . (обзор [3]). Распределение по  $Q_{2\alpha}$  для измеренных 500  $2\alpha$ -пар, в том числе 198 “белых”, указывает на ограничение  $Q_{2\alpha}({}^8\text{Be}) \leq 0.2$  МэВ. Оно включает принятое приближение сохранения вторичными фрагментами импульса родительского ядра, кинематический эллипс распада  ${}^8\text{Be}$  и разрешение угловых измерений. Ядро  ${}^8\text{Be}$  является неизменным продуктом распада  ${}^9\text{B}$ . Основное состояние ядра  ${}^9\text{B}$  выше порога  ${}^8\text{Be} + p$  на 185.1 кэВ, а его ширина  $0.54 \pm 0.21$  кэВ [12]. При изучении когерентной диссоциации изотопа  ${}^{10}\text{C}$  при энергии 1.2 А ГэВ канал  $2\text{He}+2\text{H}$  составил 82% “белых” звезд, среди которых условию  $Q_{2\alpha p}({}^9\text{B}) < 0.5$  МэВ отвечает  $(30 \pm 4)\%$ . Во всех этих  $2\alpha p$ -тройках установлена последовательность распадов  ${}^9\text{B} \rightarrow {}^8\text{Be} + p$  и  ${}^8\text{Be} \rightarrow 2\alpha$ .

События фрагментации ядра  ${}^9\text{Be}$  с образованием двух  $\alpha$ -частиц в переднем конусе фрагментации можно рассмотреть на рисунке 2.3, на нем изображено 3 различных события. Первое изображение типа “белая звезда”, второе звезда с протоном отдачи (g-частица) и третье изображение — это событие с тяжелым осколком (b-частица). На рисунке 2.4 изображено событие фрагментации ядра  ${}^9\text{Be}$  с образованием 2 b частиц, 2 g частиц в широком конусе.

#### «белая» звезда



#### звезда с протоном отдачи (g-частицей)



#### звезда с тяжелым осколком (b-частицей)

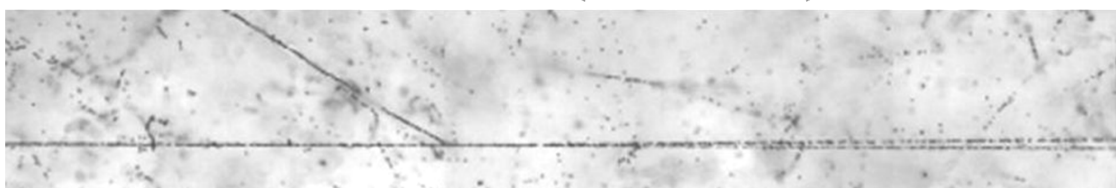


Рисунок 2.3 – События фрагментации ядра  ${}^9\text{Be}$  с образованием двух  $\alpha$ -частиц в переднем конусе фрагментации



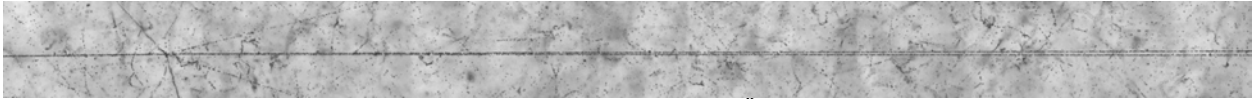


Рисунок 2.4 – Событие фрагментации ядра  ${}^9\text{Be}$  с образованием 2 b частиц, 2 g частиц в широком конусе

## 2.2 Фрагментация ${}^{10}\text{B} \rightarrow 2\text{He}(+H)$

В исследовании диссоциации релятивистского ядра  ${}^6\text{Li}$  была детально продемонстрирована кластеризация  $\alpha + d$ , информация которой представляет ценность для выяснения роли более сложной кластерной структуры  $2\alpha + d$  в ядре –  ${}^{10}\text{B}$ . Ядро  ${}^{10}\text{B}$  имеет спин равный 3, четность положительная. По представлениям оболочечной модели его структура имеет вид  $(1s)^4(1p3/2)^6$ , т.е. четыре нуклона заполняют 1s-оболочку, а следующая оболочка заполнена не полностью, так как для нее полное число нуклонов равно восьми. Ядро  ${}^{10}\text{B}$  представляет собой одно из не многих стабильных нечетно-нечетных ядер, распространенность которого в природе не превышает 20%. Размеры ядра  ${}^{10}\text{B}$  ( $R_{\text{rms}} = 2.20 \pm 0.02$  ферми) не позволяют его отнести к числу экзотических ядер, а величина его спина ( $J_\pi = 3^+$ ) соответствует оболочечной модели. Ядро  ${}^{10}\text{B}$  можно представить как  ${}^8\text{Be} \rightarrow 2\text{He}$  или  ${}^8\text{Be} \rightarrow 2H + H + n$ . Согласно исследованиям [13] известно, что эти каналы фрагментации имеют большие вероятности их наблюдения в сравнении с другими каналами. Распад  ${}^8\text{Be} \rightarrow 2\text{He}$  может происходить только в состояниях  $0^+$  и  $2^+$ . Установлено, что основным состоянием ядра  ${}^8\text{Be}$  является состояние  $0^+$ , а первое возбужденное состояние  $2^+$ .

Пороги отделения нуклонов и легчайших ядер для ядра  ${}^{10}\text{B}$  кластеризацией составляют: 4.5 МэВ ( ${}^6\text{Li} + \alpha$ ), 6.0 МэВ ( ${}^8\text{Be} + d$ ), 6.6 МэВ ( ${}^9\text{Be} + p$ ), 17.8 МэВ ( ${}^7\text{Li} + 3\text{He}$ ) и 18.7 МэВ ( ${}^7\text{Be} + t$ ) (рисунок 2.5). Удаление одного протона с p-оболочки (8.4 МэВ) ведет к образованию ядра  ${}^9\text{Be}$  со спином  $3/2$ . Зеркальной альтернативой этого канала является канал с отделением нейтрона  ${}^9\text{B} + n$ , который должен иметь ту же вероятность. Возможно, что, как и в случае ядра  ${}^{10}\text{C}$ ,  ${}^9\text{B}$  может проявиться как единственный источник ядра  ${}^8\text{Be}$  в основном состоянии  $0^+$ . В свою очередь источником ядер  ${}^8\text{Be}$  в первом возбужденном состоянии  $2^+$  в ядре  ${}^{10}\text{B}$  могла бы служить кластерная конфигурация с участием дейтрона  ${}^8\text{Be}_{(2^+)} + d_{(1^+)}$ .

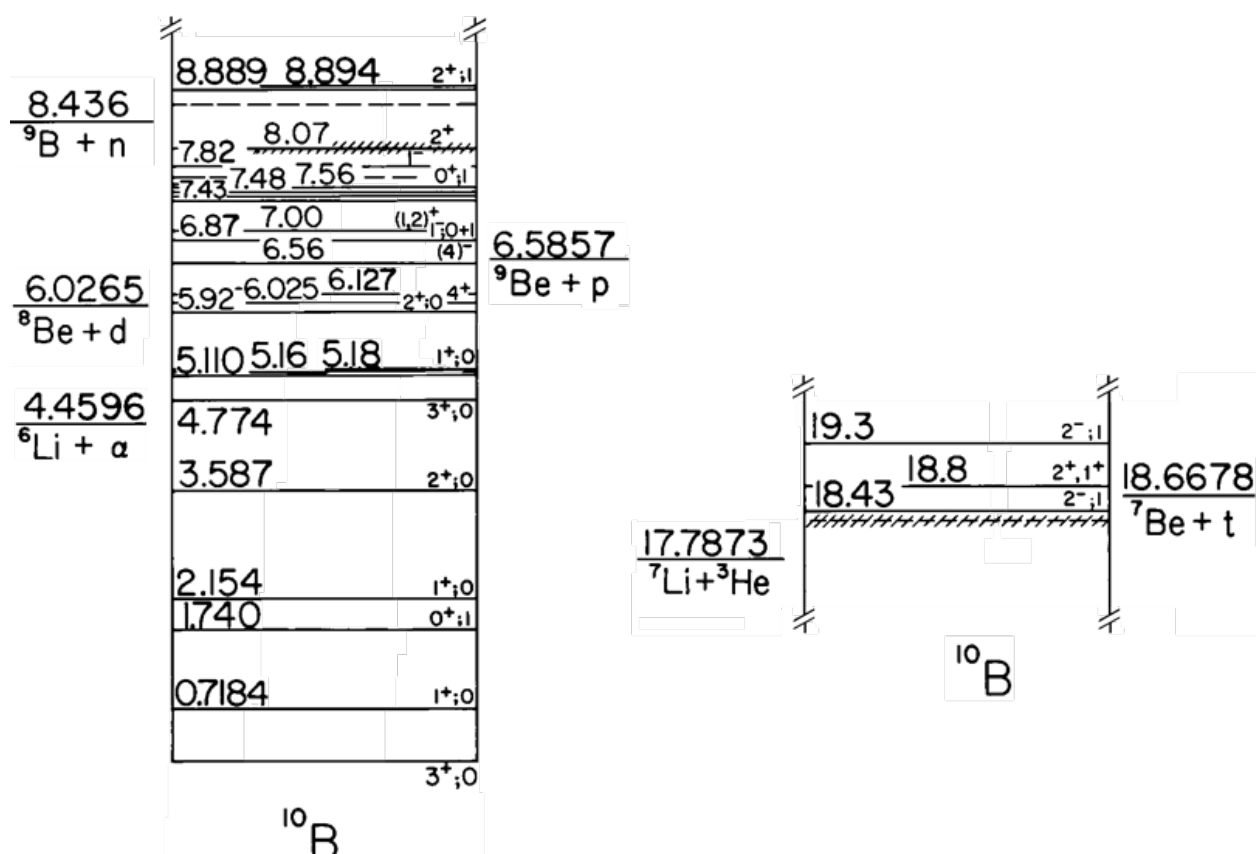


Рисунок 2.5 – Энергетические уровни возбуждения ядра  $^{10}\text{B}$ . Указаны пороги отделения нуклонов и легчайших ядер

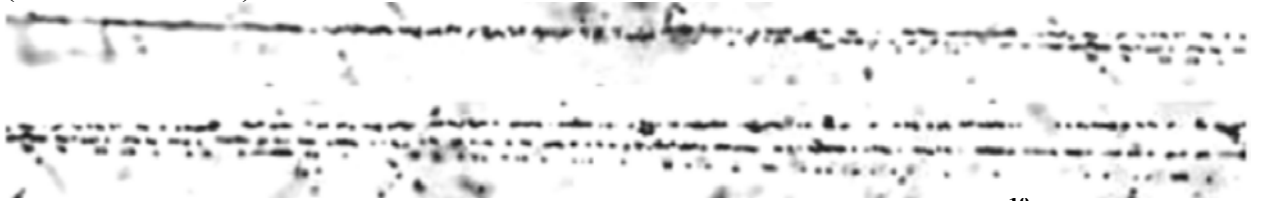
Облучение ЯЭ ядрами  $^{10}\text{B}$  с энергией 1 А ГэВ было выполнено в одном из первых сеансов на нуклотроне ОИЯИ. При анализе событий когерентной диссоциации проявилось доминирование “белых” звезд в канале  $2\text{He} + \text{H}$  при их подавлении в канале  $\text{Be} + \text{H}$  (не более 2%). Обнаружение нестабильного ядра  $^9\text{B}$  в структуре радиоактивного ядра  $^{10}\text{C}$  указало на возможность диссоциации ядра  $^{10}\text{B}$  с участием  $^9\text{B}$ . Также изучение диссоциации ядра  $^9\text{Be}$  в ЯЭ позволяет интерпретировать структуру ядра  $^{10}\text{B}$  с учетом возможного присутствия в нем суперпозиции виртуальных состояний  $^8\text{Be}_{g.s.}$  и  $^8\text{Be}_{2+}$ .

При просмотре по следам пучковых ядер  $^{10}\text{B}$  на длине 241 м было найдено 1664 ядерные звезды. Распределение найденных среди них 127 “белых” звезд  $^{10}\text{B}$  подтверждает лидерование канала  $2\text{He} + \text{H}$  (78%) и подавление канала  $\text{Be} + \text{H}$  (1%), который должен соответствовать конфигурации  $^9\text{Be} + \text{p}$ . Статистика событий в остальных каналах распределилась следующим образом:  $\text{He} + 3\text{H}$  — 12%,  $\text{Li} + \text{He}$  — 4% и  $\text{Li} + \text{He}$  — 4%.

В измерениях по углу разлета  $\Theta_{2\text{He}}$  статистика пар  $2\text{He}$  в интервале  $0 < \Theta_{2\text{He}} < 10.5$  мрад, соответствующем распаду  $^8\text{Be}_{g.s.}$ , охватывает  $(25 \pm 5)\%$  “белых” звезд  $^{10}\text{B} \rightarrow 2\text{He} + \text{H}$ . В интервале углов разлета  $\Theta(^8\text{Be}_{g.s.} + \text{H}) < 25$  мрад, соответствующем распадам  $^9\text{B}$ , находятся только  $(14 \pm 3)\%$  “белых” звезд  $^{10}\text{B} \rightarrow 2\text{He} + \text{H}$ . Таким образом, распадами  $^9\text{B}$  объясняется только  $(56 \pm 16)\%$  распадов  $^8\text{Be}_{g.s.}$ . Стоит подчеркнуть отличие этого вывода от случая  $^{10}\text{C}$ , где это соответствие было полным. Тем самым получает основание

предположение о существовании в ядре  $^{10}\text{B}$  суперпозиции основ  $^8\text{Be}_{g.s.} / ^8\text{Be}_{2+}$  наряду с  $^9\text{B}$ , проявляющейся в виде избытка событий с  $^8\text{Be}_{g.s.}$ .

Статистика “белых” звезд  $^{10}\text{B}$ , найденных без выборки, позволяет оценить соотношение вероятностей каналов  $^9\text{B}+n$  и  $^9\text{Be}+p$  как  $6 \pm 1$ , что кажется неожиданным, на рисунке 2.6 изображено событие фрагментации ядра  $^{10}\text{B} \rightarrow 2\text{He}+n$ . Альтернативное объяснение асимметрии вероятностей зеркальных каналов могло бы основываться на качественно более широком пространственном распределении нейтронов по сравнению с протонами. Однако этот вариант представляется маловероятным, поскольку неупругое сечение взаимодействия релятивистских ядер  $^{10}\text{B}$  не имеет усиленного (экзотического) поведения.



**Рисунок 2.6 – Макрофотография события фрагментации ядра  $^{10}\text{B} \rightarrow 2\text{He}+n$  без разрушения ядра-мишени**

Возможно, что этот факт указывает на присутствие основы  $^9\text{Be}$  в  $^{10}\text{B}$  преимущественно в форме суперпозиции  $^8\text{Be}_{2+} / ^8\text{Be}_{g.s.} + n$  (ядерной молекулы), что проявляется в диссоциации с прямым образованием состояний  $^8\text{Be}_{g.s.}$  (без распадов  $^9\text{B}$ ) и  $^8\text{Be}_{2+}$ . Основа же в виде  $^9\text{B}$  представляет собой изначально такую кластерную форму.

### 2.3 Сравнение и резюме

Проведен сравнительный анализ топологии фрагментации ядер бериллия и бора. Выявлено, что при распаде ядра  $^9\text{Be}$  наиболее часто образуются два фрагмента ядер He. При этом пары фрагментов ядер He характеризуются тем, что в их угловых распределениях имеются два пика в области малых и больших относительных углов. Эти пики также заметны в распределениях по инвариантной массе пар альфа-частиц, где первый пик соответствует распадам ядра  $^8\text{Be}$  из основного состояния, а второй пик распадам ядра  $^8\text{Be}$  из первого возбужденного состояния. При этом эти пики имеют одинаковые веса. Что касается диссоциации релятивистских ядер  $^{10}\text{B}$ , то здесь также превалирует канал с образованием пары фрагментов ядер He и одного нуклона (нейтрона или протона). Кластерная структура ядер  $^9\text{Be}$  и  $^{10}\text{B}$  различается на один протон. В распределениях по парным углам и инвариантной массе альфа-частиц также видны события с распадами  $^8\text{Be}$  из основного состояния. Эти распады происходят как в прямых реакциях ( $^{10}\text{B} \rightarrow ^8\text{Be} + d(np) \rightarrow 2\text{He} + d(np)$ ), так и в событиях с распадом нестабильного ядра  $^9\text{B}$  ( $^{10}\text{B} \rightarrow ^9\text{B} + n \rightarrow ^8\text{Be} + p + n \rightarrow 2\text{He} + p + n$ ). Следует отметить, что канал  $^9\text{B} + n$  в топологии соответствует зеркальному ему каналу  $^9\text{Be} + p$  в отношении 6/1. Такая асимметрия вероятностей зеркальных каналов могла бы объясняться качественно более широким пространственным распределением нейтронов по сравнению с протонами. Однако этот вариант

кажется маловероятным, поскольку неупругое сечение взаимодействия релятивистских ядер  $^{10}\text{B}$  не демонстрирует усиленного (экзотического) поведения. Возможно, что этот факт свидетельствует о том, что основа  $^9\text{Be}$  в  $^{10}\text{B}$  присутствует преимущественно в форме суперпозиции  $^8\text{Be} (2+) / ^8\text{Be} (g.s.) + n$  (ядерной молекулы), что проявляется в диссоциации с прямым образованием состояний  $^8\text{Be} (g.s.)$  (без распадов  $^9\text{B}$ ) и  $^8\text{Be} (2+)$ . Основа же в виде  $^9\text{B}$  представляет собой изначально такую кластерную форму.

### 3 Фрагментация ядер мишени под действием релятивистских легких ядер

#### 3.1 Особенности фрагментации ядер мишени в событиях ${}^9\text{Be} + \text{Em}$ и ${}^{10}\text{B} + \text{Em}$

Методом ядерных фотоэмульсий вершина взаимодействия налетающего ядра с ядром из состава эмульсии реконструируется с высокой степенью точности. В данной работе было выделено два типа ядро-ядерных взаимодействий: события с явным наблюдением треков фрагментов ядра-мишени, рожденных мезонов и фрагментов налетающих ядер. Вторым типом событий так называемые “белые” звезды, в которых, где треки рожденных мезонов и фрагментов мишени не наблюдаются. Числом фрагментов мишени в конечном состоянии в событии будут называться множественность фрагментов мишени и обозначаться символом  $nh$ . В свою очередь фрагменты мишени разделяются на серые “g” и черные “b” фрагменты мишени. Пробег b-частиц составляет порядка 3 мм, а длина трека g-частицы более 3 мм. Идентификация треков s-частиц проводилась визуальным способом, поскольку плотность проявленных зерен вдоль трека существенно ниже, чем для треков других частиц. Таким образом определялась множественность фрагментов мишени в событии. В таблицах А.1-2, Б.1-2, В.1-2 представлены распределения по множественности вышеуказанных частиц. Распад ядер мишени выглядит следующим образом рисунки 3.1, 3.2. Есть белые звезды, в которых мишени видимым образом не распадаются и нет рожденных мезонов в узком конусе. При наблюдении разрушения ядра мишени и при этом в этом событии наблюдается образование ядер  ${}^8\text{Be}$ , топология фрагментов ядер мишени будет рассмотрена ниже.

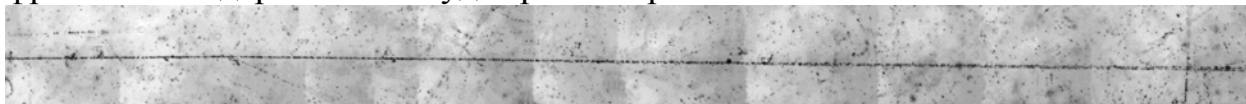


Рисунок 3.1 – Диссоциация ядра  ${}^{10}\text{B}$  с образованием двух фрагментов ядер He и пары мезонов

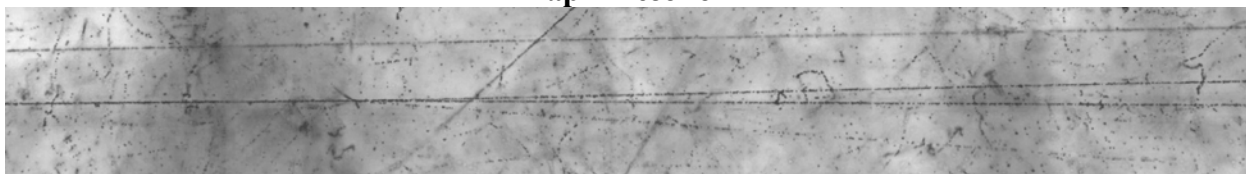


Рисунок 3.2 – Диссоциация ядра  ${}^9\text{Be}$  с образованием трех тяжелых фрагментов, одного протона отдачи и одного фрагмента He:  ${}^9\text{Be} \rightarrow 2\text{He} + s + 3b + 1g$

Для анализа фрагментации ядер-мишеней из состава ядерной фотоэмульсии были изучены события диссоциации релятивистских ядер  ${}^{10}\text{B}$  и  ${}^9\text{Be}$ . Наибольшей статистической обеспеченностью во фрагментации ядер  ${}^{10}\text{B}$  являются каналы:  ${}^{10}\text{B} \rightarrow 2\text{He} + \text{H}$  - 173 события и без наблюдения трека ядра  $\text{H}$  - 107 событий. Во фрагментации ядер  ${}^9\text{Be}$  таким каналом является  ${}^9\text{Be} \rightarrow 2\text{He}$  - 475 событий. В ходе пред магистерской практики статистика событий вышеуказанных событий была дополнена 20 и 25 событиями в соответствующих каналах. Для этого использовался просмотровый микроскоп МБИ-9, с суммарным оптическим увеличением 600 крат. Сканирование образцов ядерной эмульсии проводился методом просмотра

вдоль трека. Стоит отметить, что данный метод просмотра является время затратной процедурой. Однако, он позволяет вести полноценный анализ топологии фрагментации ядер без ограничений. Таким образом, окончательная статистика событий составила:  $^{10}\text{B} \rightarrow 2\text{He} + \text{H}$  - 193 события,  $^{10}\text{B} \rightarrow 2\text{He}$  - 107 событий и  $^9\text{Be} \rightarrow 2\text{He}$  - 500 событий.

Возможности программного пакета ROOT, который был использован для анализа данных о множественности сильно ионизирующих и ливневых частиц, полученных в результате взаимодействия ядер бора и бериллия с ядрами эмульсии, велики. ROOT — это открытый фреймворк для научного анализа и визуализации больших объемов данных, который широко используется в физике высоких энергий и других областях. ROOT позволяет работать с данными в различных форматах, создавать графики, гистограммы, функции, кривые и другие объекты, а также использовать мощный интерпретатор C++, идеальный для быстрого прототипирования. Кроме того, ROOT обладает уникальной динамической и мощной связью Python  $\rightleftharpoons$  C++, которая позволяет использовать Python для работы с данными и объектами ROOT [14].

Для построения распределений по множественности сильно ионизирующих частиц  $nh=ng+nb$ , а также для ливневых частиц  $ns$  был использован класс TH1F, который представляет собой одномерную гистограмму с плавающими значениями. Были заданы параметры гистограммы, такие как название, оси, количество и диапазон бинов (рисунок 3.3).

```
TH1F* h1=new TH1F("nb", "Histogram title; nb; Counts", 8, 0, 8);
TH1F* h2=new TH1F("ng", "Histogram title; ng; Counts", 8, 0, 8);
TH1F* h3=new TH1F("ns", "Histogram title; ns; Counts", 8, 0, 8);
TH1F *h4 = new TH1F("nh", "Histogram title; nh; Counts", 12, 0, 12);
```

**Рисунок 3.3 – Пример задания параметров гистограмме**

Затем гистограмма была заполнена данными из файла B-01-18.dat, который содержит информацию о номере события, количестве b, g и s частиц, количестве фрагментов He и H в конечном состоянии. В случае бора имелось 2 канала фрагментации  $^{10}\text{B} \rightarrow 2\text{He} + \text{H}$ ,  $^{10}\text{B} \rightarrow 2\text{He}$ . То есть в первом случае была сделана выборка для событий с количеством He равным 2 и H равным 1, во втором случае аналогично два гелия, но при этом ноль фрагментов H (рис 3.4). В случае бериллия выборку делать не пришлось.

```

if (F)
{
    while(!F.eof())
    {
        F >>name>>nb>>ng>>ns>>n4>>n3>>n2>>n1;
        F.ignore(1024, '\n');
        if (!F.good()) break;
        if (n2==2 && n1==1){
            h1->Fill(nb);
            h2->Fill(ng);
            h3->Fill(ns);
            if (nb+ng>=0)
            {
                h4->Fill(nb+ng);
            }
        }
    }
    F.close();
}
else cout<<"No file"<<endl;

```

```

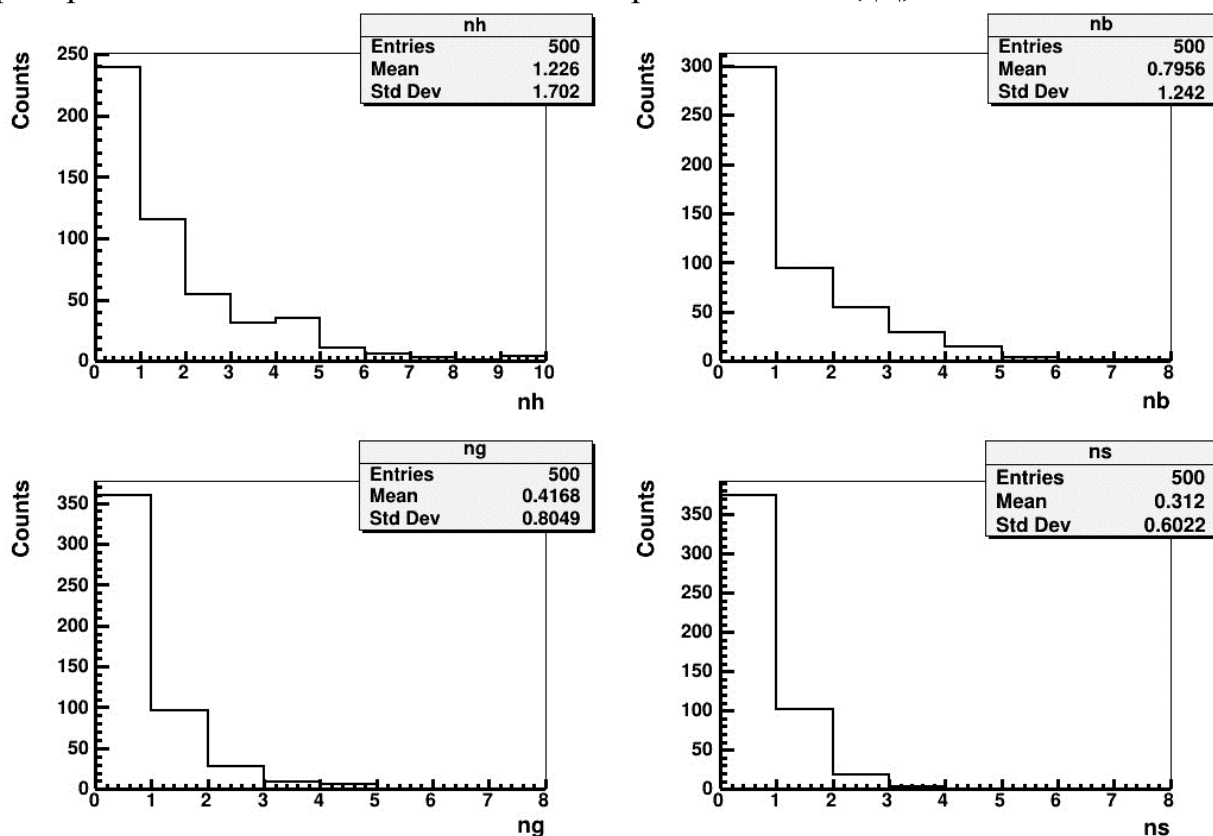
if (F)
{
    while(!F.eof())
    {
        F >>name>>nb>>ng>>ns>>n4>>n3>>n2>>n1;
        F.ignore(1024, '\n');
        if (!F.good()) break;
        if (n2==2 && n1==1){
            h1->Fill(nb);
            h2->Fill(ng);
            h3->Fill(ns);
            if (nb+ng>=0)
            {
                h4->Fill(nb+ng);
            }
        }
    }
    F.close();
}
else cout<<"No file"<<endl;

```

**Рисунок 3.4 –Заполнение гистограмм данными из файла, выборка по He и H**

Для каждого такого события были заполнены гистограммы nb, ng, ns и nh. Была произведена нормировка для гистограммы на единицу, чтобы получить относительные частоты. С полным кодом программ можно ознакомиться в приложениях 1-3.

Для отображения гистограмм использовался класс TCanvas, который представляет собой окно для рисования объектов ROOT. Результат можно увидеть на рисунках 3.5-10, с программным кодом для данных распределений можно ознакомиться в приложениях Г, Д, Е.



**Рисунок 3.5 – Распределение по множественности nh, nb, ng, ns ( ${}^9\text{Be}$ ), где  $nh \geq 0$**

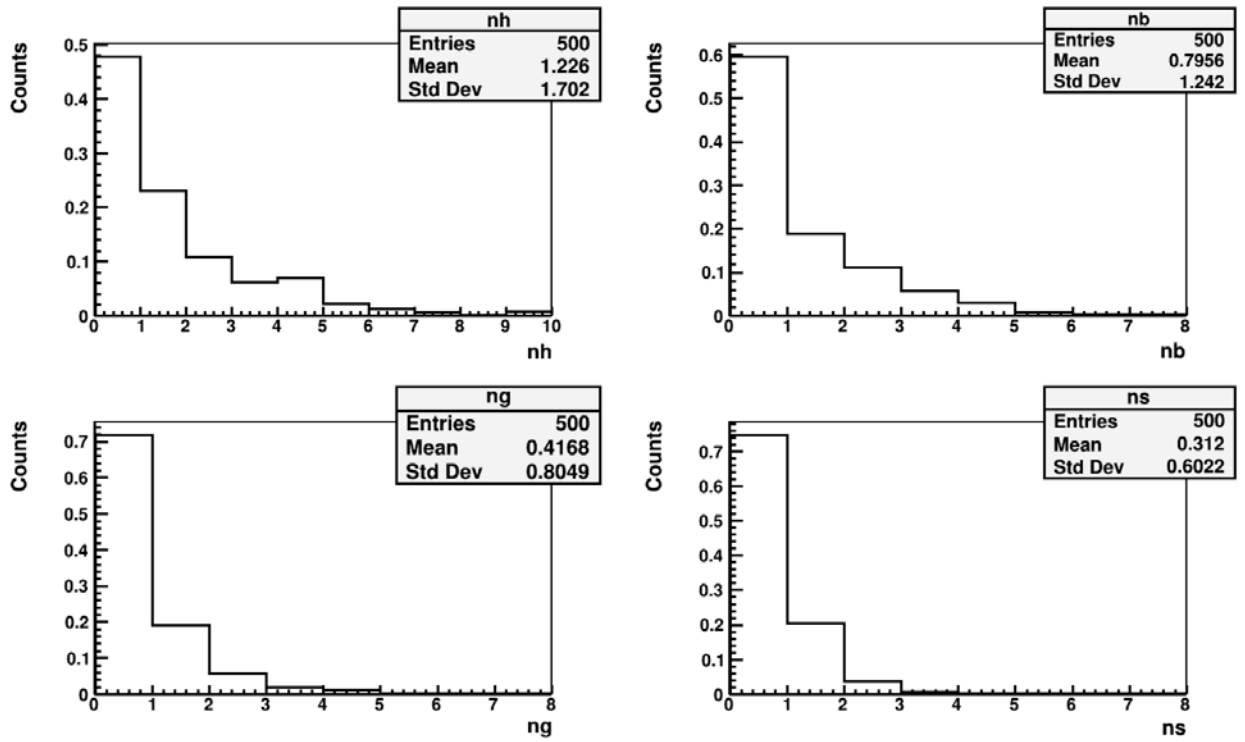


Рисунок 3.6 – Нормированное распределение по множественности  $nh$ ,  $nb$ ,  $ng$ ,  $ns$  ( ${}^9\text{Be}$ ), где  $nh \geq 0$

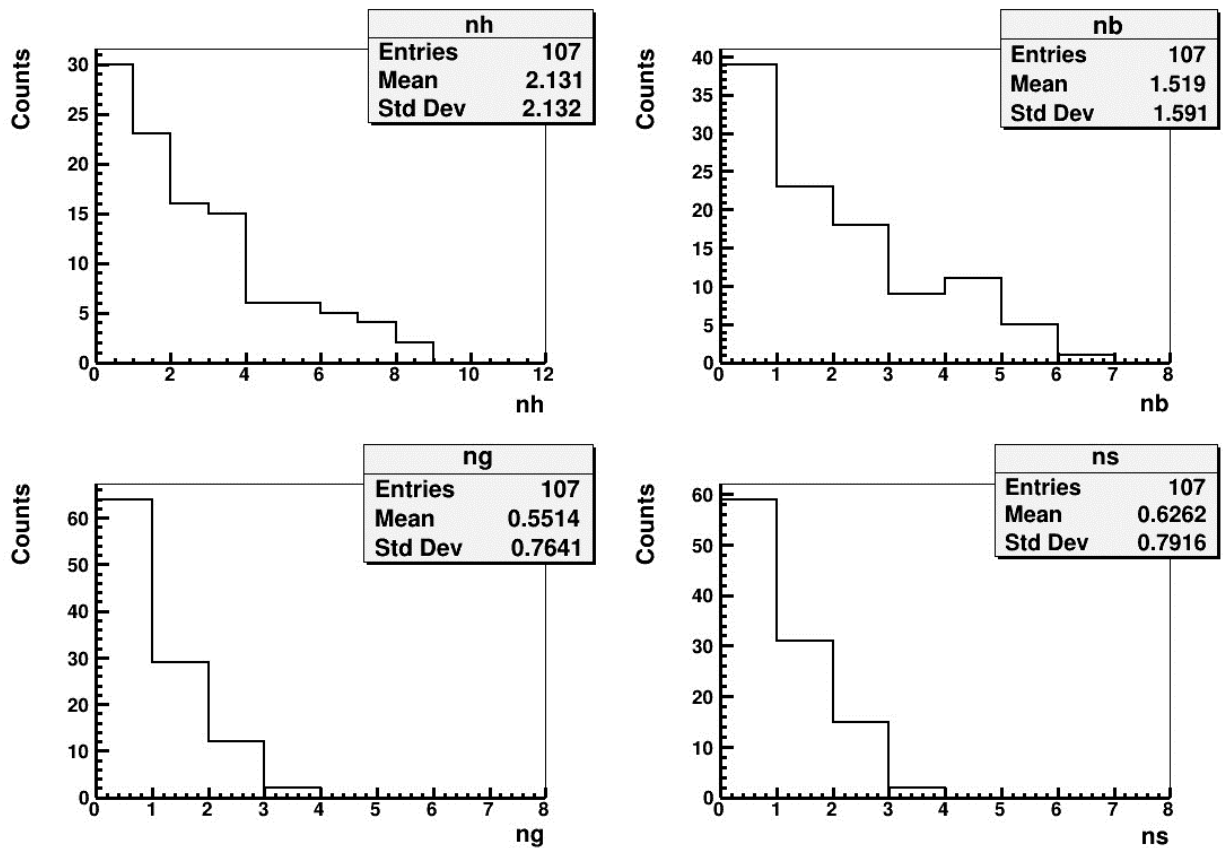


Рисунок 3.7 – Распределение по множественности  $nh$ ,  $nb$ ,  $ng$ ,  $ns$  ( ${}^{10}\text{B} \rightarrow 2\text{He}$ ), где  $nh \geq 0$



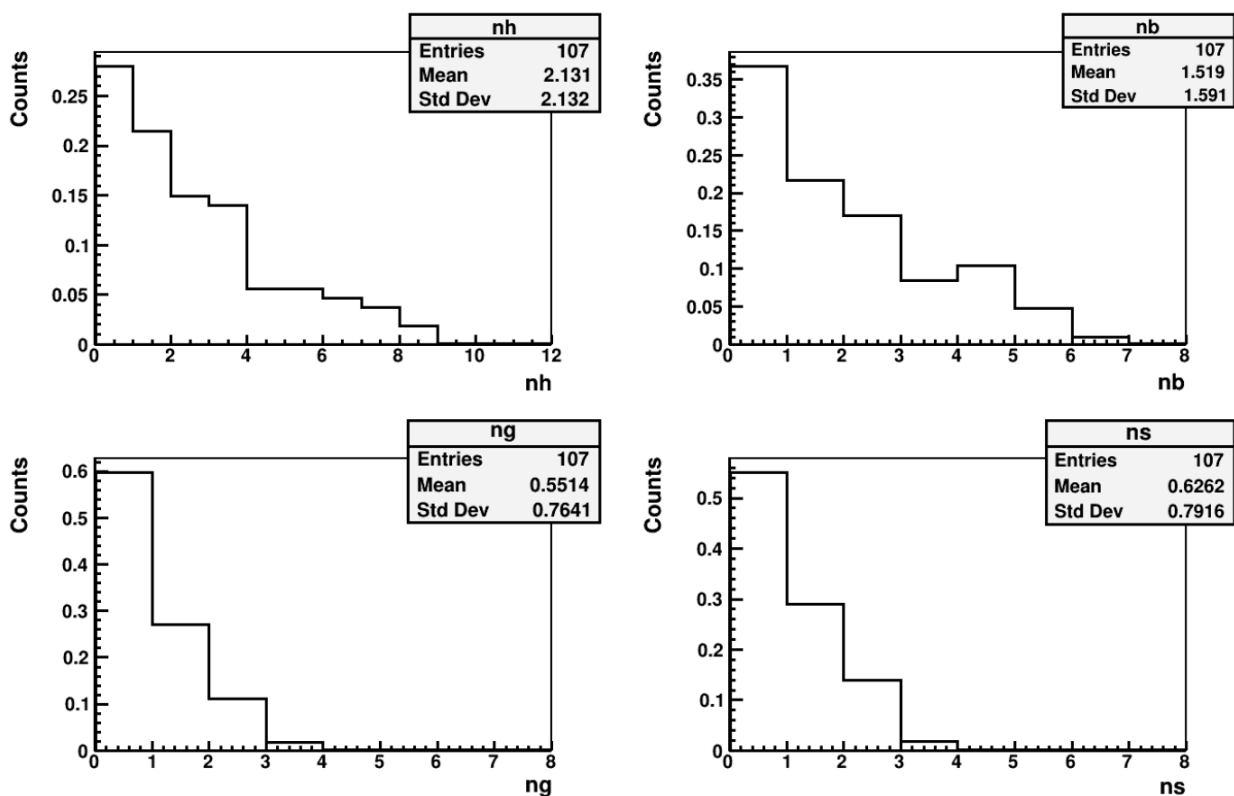


Рисунок 3.8 – Нормированное распределение по множественности  $nh$ ,  $nb$ ,  $ng$ ,  $ns$  ( $^{10}\text{B} \rightarrow 2\text{He}$ ), где  $nh \geq 0$

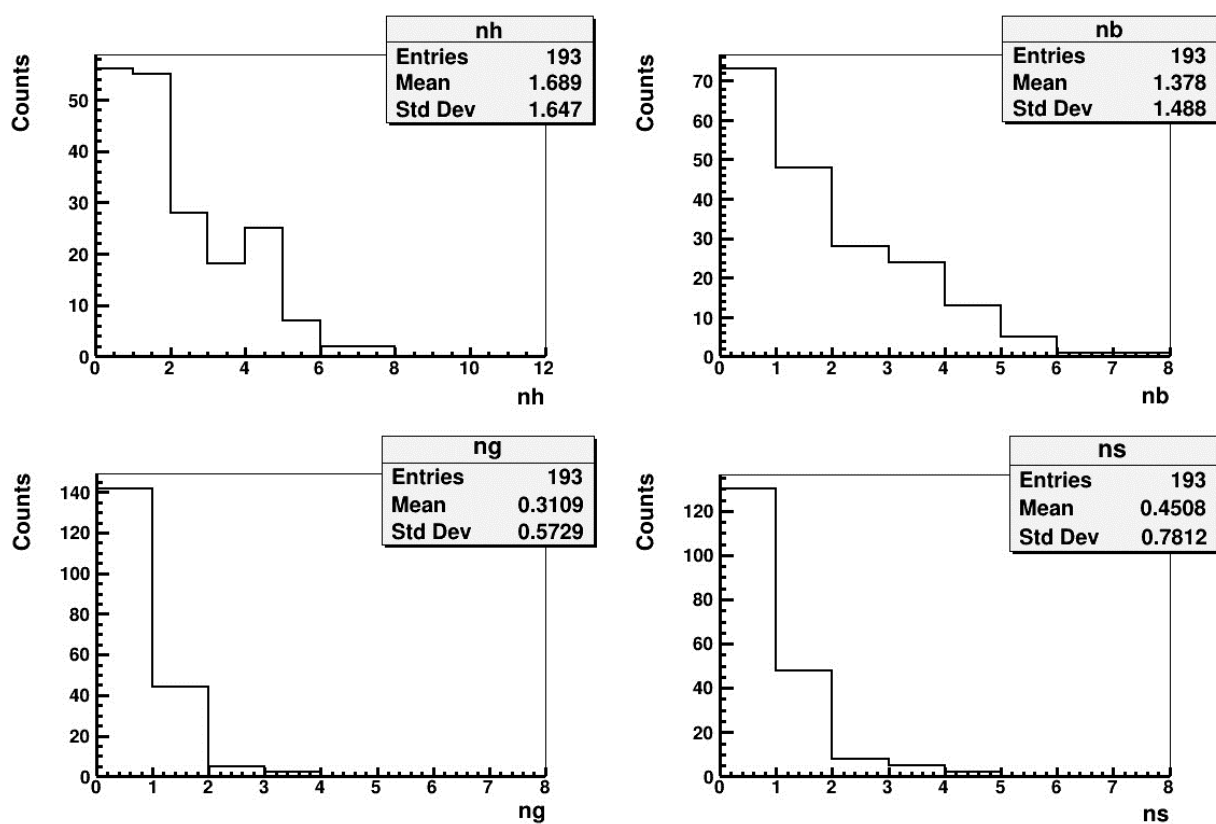


Рисунок 3.9 – Распределение по множественности  $nh$ ,  $nb$ ,  $ng$ ,  $ns$  ( $^{10}\text{B} \rightarrow 2\text{He} + \text{H}$ ), где  $nh \geq 0$

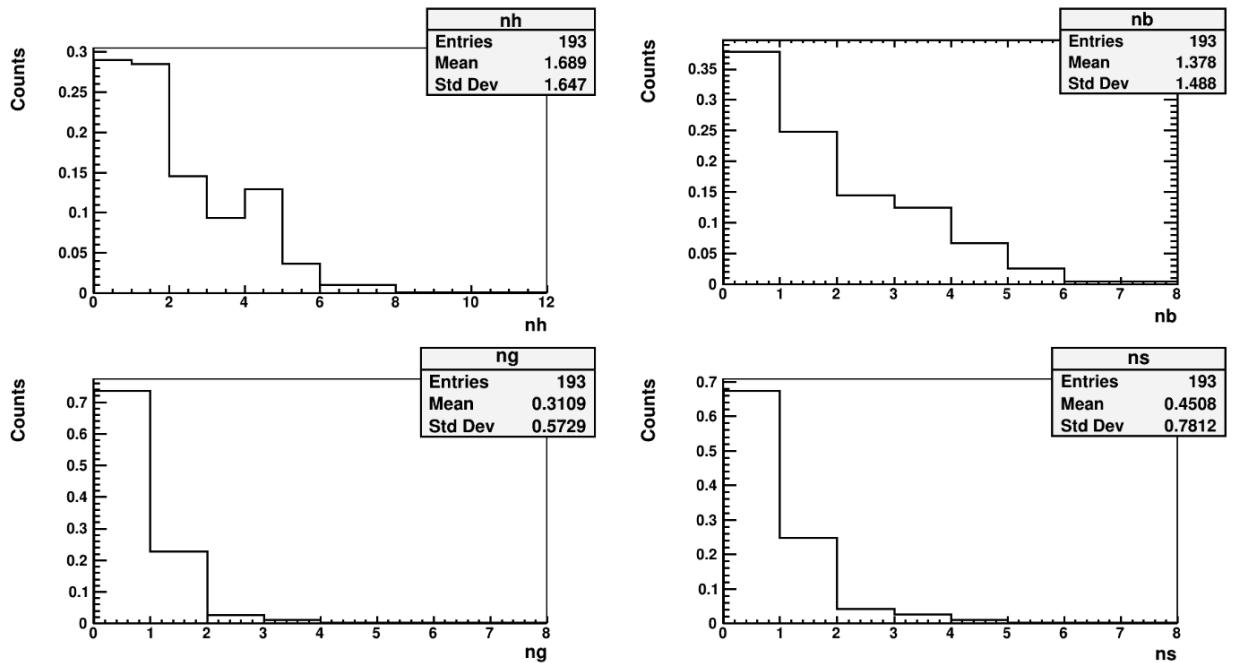


Рисунок 3.10 – Нормированное распределение по множественности nh, nb, ng, ns ( $^{10}\text{B} \rightarrow 2\text{He} + \text{H}$ ), где  $nh \geq 0$

Из гистограмм видно, что распределения имеют пиковый характер и убывают экспоненциально при увеличении множественности. В группу с  $nh < 6$  входят взаимодействия на ядрах H и группы ядер CNO. А в группу с  $nh \geq 6$  образуют события взаимодействий на тяжелых ядрах из состава ядерной эмульсии Ag и Br. В таблице 2 представлены средние множественности b, g и s частиц в событиях  $^{10}\text{B} \rightarrow 2\text{He} + \text{H}$  и  $^9\text{Be} \rightarrow 2\text{He}$ . Стоит заметить, что вклад событий с  $nh \geq 6$  во всех случаях явно подавлен.

В таблице 2 приведены средние множественности фрагментов мишени. b-частицы в основном рождаются при возбуждении спектатора мишени. Фрагменты g-частиц рождаются в каскадных соударениях в спектаторе мишени и партисипанта. Вклад спектаторов мишени больше, чем партисипантов мишени. При низких энергиях вклад партисипантов увеличивается, а спектаторный вклад уменьшается за счет эффекта торможения. Это означает, что события с большим числом pg имеют высокую вероятность для наблюдения.

Таблица 2 – Средние множественности частиц для событий с  $nh \geq 0$ .

	$\langle nh \rangle$	$\langle nb \rangle$	$\langle ng \rangle$	$\langle ns \rangle$	$N_{\text{count}}$
$^{10}\text{B} \rightarrow 2\text{He} + \text{H}$	$1.7 \pm 0.1$ RMS 1.6	$1.4 \pm 0.1$ RMS 1.5	$0.3 \pm 0.04$ RMS 0.6	$0.5 \pm 0.05$ RMS 0.8	193
$^{10}\text{B} \rightarrow 2\text{He}$	$2.1 \pm 0.2$ RMS 2.1	$1.5 \pm 0.15$ RMS 1.6	$0.6 \pm 0.07$ RMS 0.8	$0.6 \pm 0.08$ RMS 0.8	107
$^9\text{Be} \rightarrow 2\text{He}$	$1.2 \pm 0.1$ RMS 1.8	$0.8 \pm 0.05$ RMS 1.2	$0.4 \pm 0.03$ RMS 0.8	$0.3 \pm 0.03$ RMS 0.6	500

Так же были построены корреляции множественности  $\langle n_i \rangle = f(n_j)$  и их аппроксимация линейной зависимостью  $\langle n_i \rangle = a + k n_j$  от  $nh$ ,  $nb$ ,  $ng$  и  $ns$  для трех каналов фрагментации. Результат можно рассмотреть на рисунках 3.11-16.

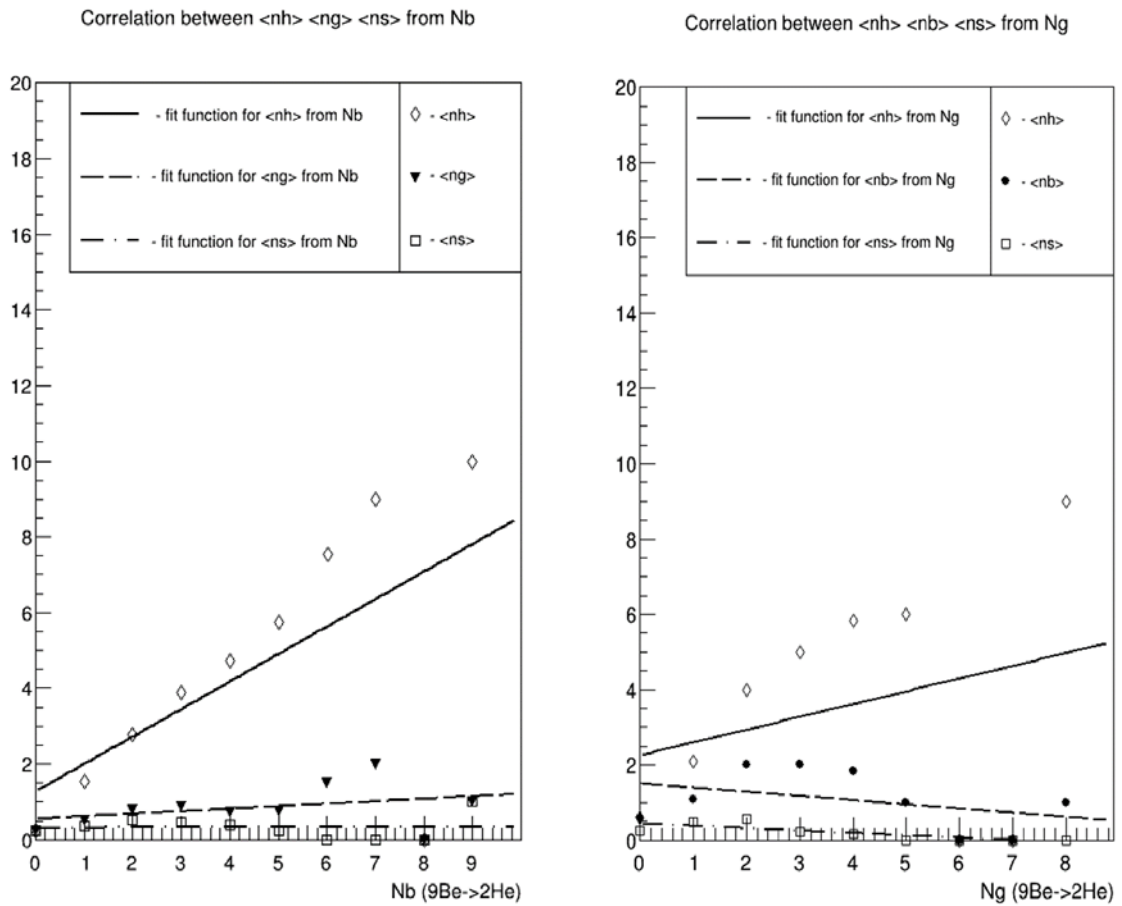


Рисунок 3.11 – Корреляция между  $\langle nh \rangle$ ,  $\langle ng \rangle$ ,  $\langle ns \rangle$  от  $nb$ ;  $\langle nh \rangle$ ,  $\langle nb \rangle$ ,  $\langle ns \rangle$  от  $ng$  для канала  ${}^9\text{Be} \rightarrow 2\text{He}$

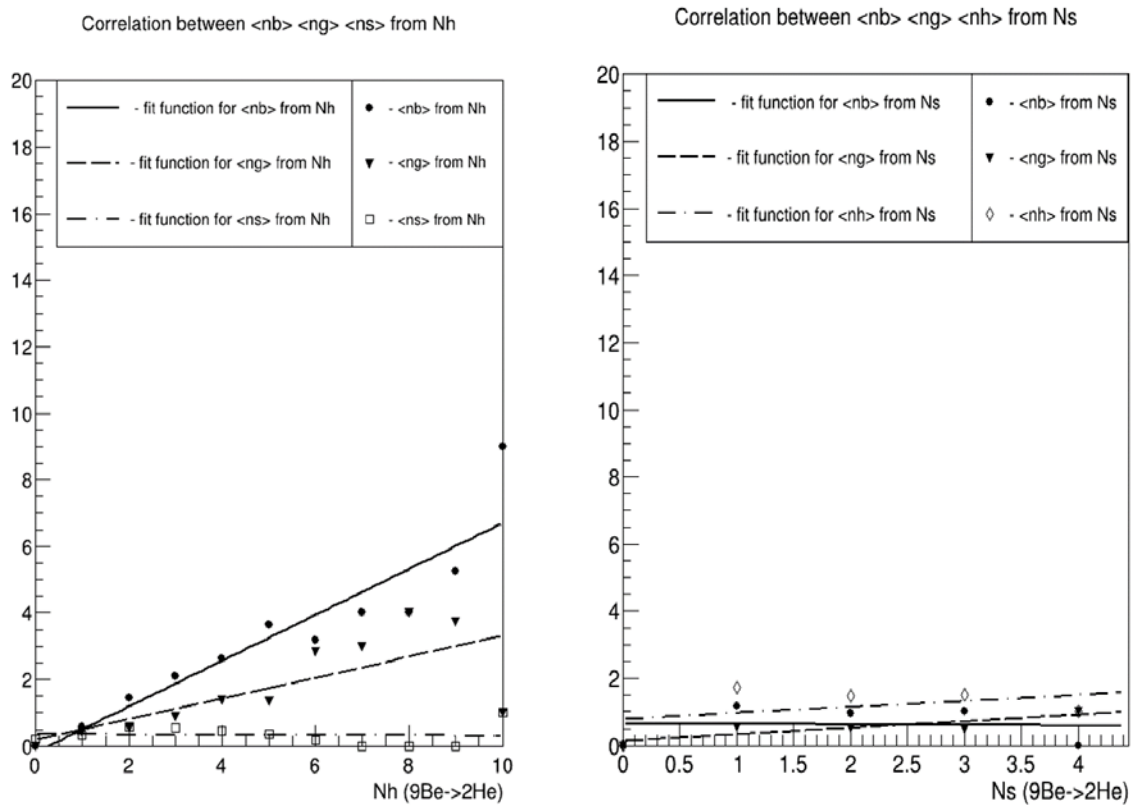


Рисунок 3.12 – Корреляция между  $\langle nb \rangle$ ,  $\langle ng \rangle$ ,  $\langle ns \rangle$  от  $nh$ ;  $\langle nh \rangle$ ,  $\langle nb \rangle$ ,  $\langle ng \rangle$  от  $ns$  для канала  ${}^9\text{Be} \rightarrow 2\text{He}$

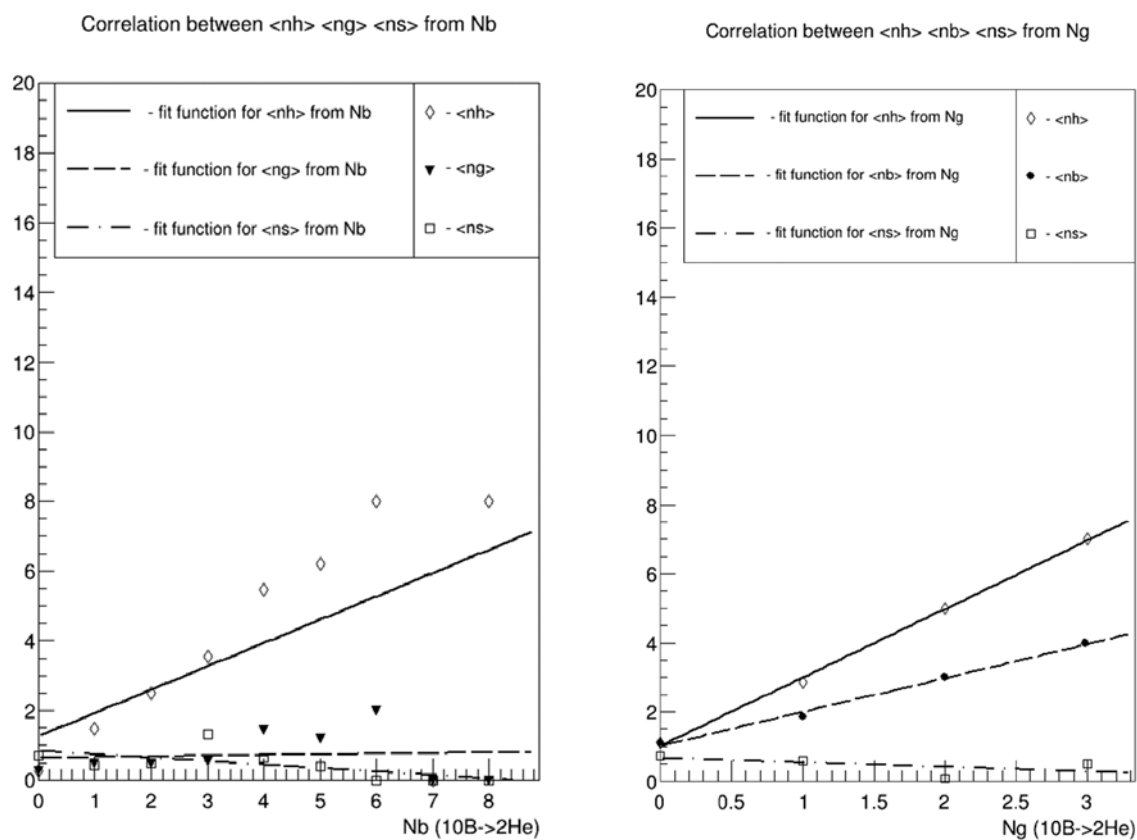


Рисунок 3.13 – Корреляция между  $\langle nh \rangle$ ,  $\langle ng \rangle$ ,  $\langle ns \rangle$  от  $nb$ ;  $\langle nh \rangle$ ,  $\langle nb \rangle$ ,  $\langle ns \rangle$  от  $ng$  для канала  ${}^{10}\text{B} \rightarrow 2\text{He}$

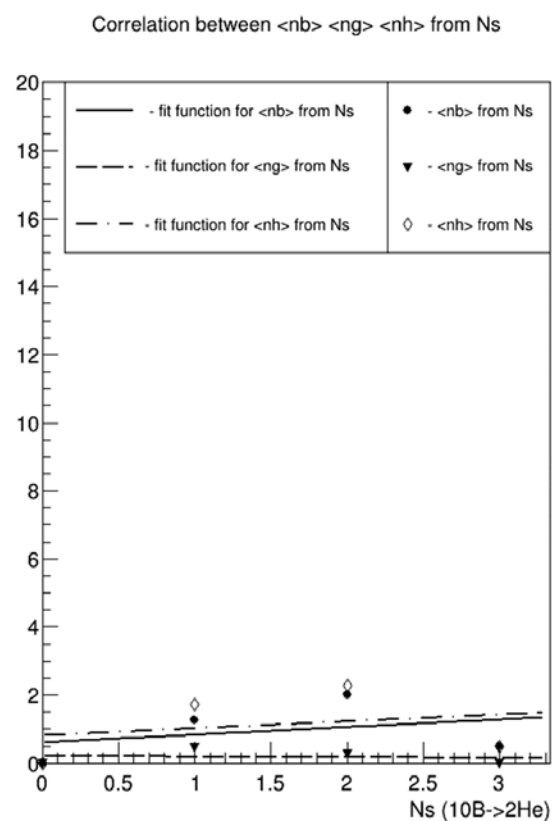
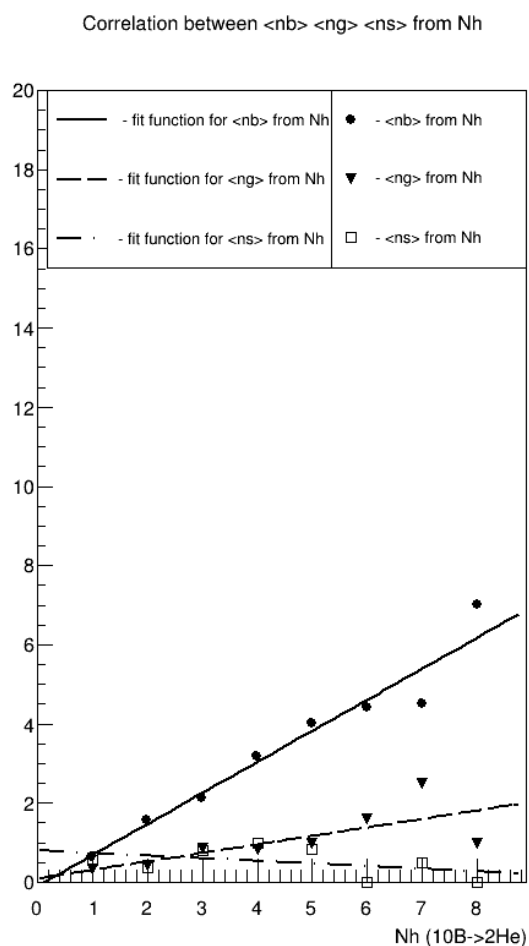


Рисунок 3.14 – Корреляция между  $\langle nb \rangle$ ,  $\langle ng \rangle$ ,  $\langle ns \rangle$  от  $nh$ ;  $\langle nh \rangle$ ,  $\langle nb \rangle$ ,  $\langle ng \rangle$  от  $ns$  для канала  $^{10}\text{B} \rightarrow 2\text{He}$

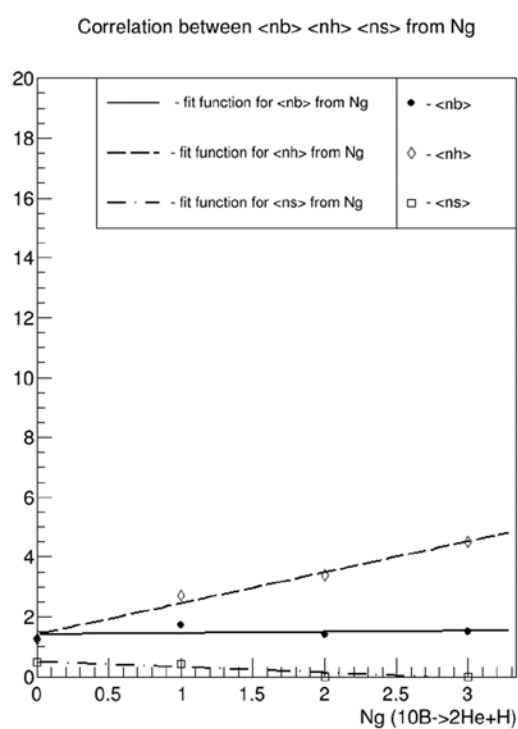
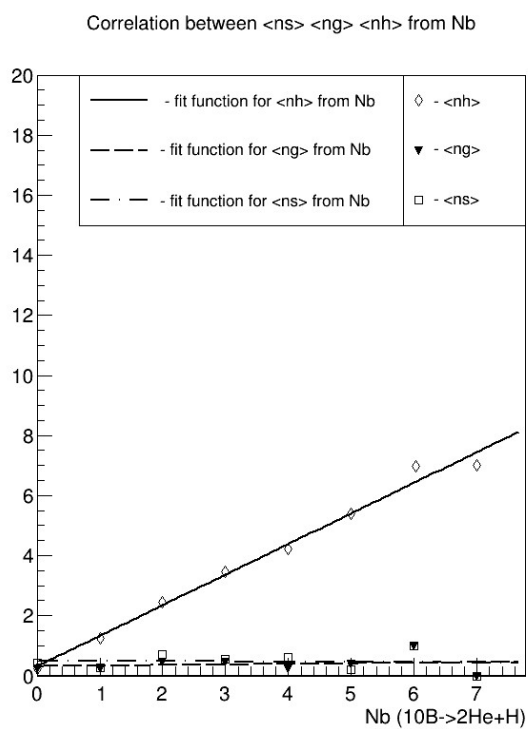


Рисунок 3.15 – Корреляция между  $\langle nh \rangle$ ,  $\langle ng \rangle$ ,  $\langle ns \rangle$  от  $nb$ ;  $\langle nh \rangle$ ,  $\langle nb \rangle$ ,  $\langle ns \rangle$  от  $ng$  для канала  $^{10}\text{B} \rightarrow 2\text{He} + \text{H}$

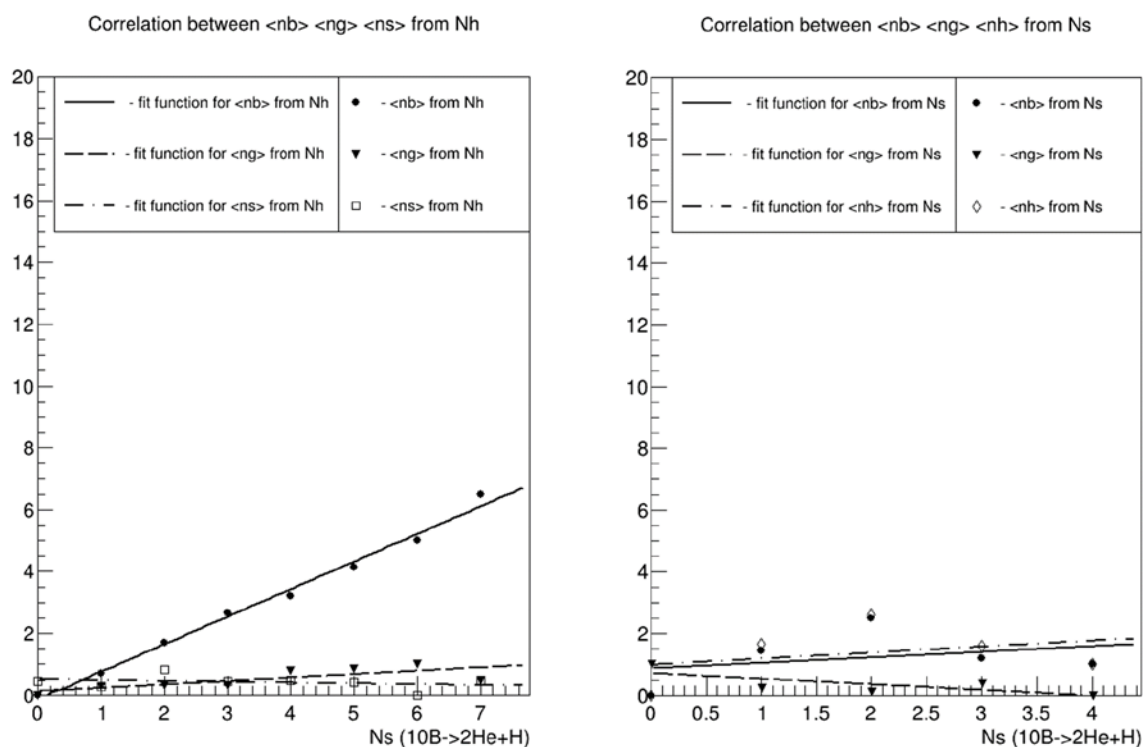


Рисунок 3.16 – Корреляция между  $\langle nb \rangle$ ,  $\langle ng \rangle$ ,  $\langle ns \rangle$  от  $nh$ ;  $\langle nh \rangle$ ,  $\langle nb \rangle$ ,  $\langle ng \rangle$  от  $ns$  для канала  $^{10}\text{B} \rightarrow 2\text{He}$

Во всех графиках наблюдается тенденция к уменьшению среднего числа s-частиц с ростом b и g - частиц. Также видно, что коэффициент k для s-частиц меньше, чем для g- и b-частиц. Данный факт может указывать на отсутствие заметного образования мезонов в событиях. В таблицах 3 – 6 представлены значения коэффициента k для трех каналов фрагментации.

На рисунке 3.12 (слева) средняя множественность  $\langle nb \rangle$  растет с числом  $nh$  до величины  $nh=5$ . Начиная с  $nh \geq 5$  отмечается некое плато (насыщение) и средняя множественность уже слабо зависит от  $nh$ , за исключением крайней точки, которая обладает малой статистической обеспеченностью. Для  $\langle ng \rangle$  наблюдается рост с ростом числа сильно ионизирующих частиц мишени. Для данных по фрагментации мишени под действием ядер  $^{10}\text{B}$  наблюдается монотонный рост как для среднего числа b-частиц, так и для g-частиц.

Таблица 3 – Коэффициенты наклона от  $nh$  для 3 каналов фрагментации

Канал фрагментации	$\langle nb \rangle$	$\langle ng \rangle$	$\langle ns \rangle$
$^{10}\text{B} \rightarrow 2\text{He} + \text{H}$	0.892 +/- 0.04	0.109 +/- 0.04	-0.026 +/- 0.04
$^{10}\text{B} \rightarrow 2\text{He}$	0.785 +/- 0.06	0.215 +/- 0.06	-0.066 +/- 0.04
$^9\text{Be} \rightarrow 2\text{He}$	0.688 +/- 0.09	0.312 +/- 0.09	-0.004 +/- 0.03

Таблица 4 – Коэффициенты наклона от  $ng$  для 3 каналов фрагментации

Канал фрагментации	$\langle nh \rangle$	$\langle nb \rangle$	$\langle ns \rangle$
$^{10}\text{B} \rightarrow 2\text{He} + \text{H}$	1.037 +/- 0.1	0.037 +/- 0.1	-0.187 +/- 0.06
$^{10}\text{B} \rightarrow 2\text{He}$	1.981 +/- 0.05	0.981 +/- 0.05	-0.125 +/- 0.13
$^9\text{Be} \rightarrow 2\text{He}$	0.339 +/- 0.42	-0.111 +/- 0.09	-0.063 +/- 0.02

Таблица 5 – Коэффициенты наклона от  $nb$  для 3 каналов фрагментации

Канал фрагментации	$\langle nh \rangle$	$\langle ng \rangle$	$\langle ns \rangle$
$^{10}\text{B} \rightarrow 2\text{He} + \text{H}$	1.02 +/- 0.05	0.019 +/- 0.05	-0.009 +/- 0.05
$^{10}\text{B} \rightarrow 2\text{He}$	0.67 +/- 0.35	0.02 +/- 0.09	-0.1 +/- 0.05
$^9\text{Be} \rightarrow 2\text{He}$	0.727 +/- 0.32	0.066 +/- 0.06	0.002 +/- 0.04

Таблица 6 – Коэффициенты наклона от  $ns$  для 3 каналов фрагментации

Канал фрагментации	$\langle nh \rangle$	$\langle nb \rangle$	$\langle ng \rangle$
$^{10}\text{B} \rightarrow 2\text{He} + \text{H}$	0.193 +/- 0.33	0.176 +/- 0.31	-0.183 +/- 0.09
$^{10}\text{B} \rightarrow 2\text{He}$	0.206 +/- 0.56	0.224 +/- 0.45	-0.018 +/- 0.12
$^9\text{Be} \rightarrow 2\text{He}$	0.178 +/- 0.23	-0.016 +/- 0.21	0.194 +/- 0.06

### 3.2 Представление результатов с помощью ROOT

В этом подразделе разберем, как считались средние значения вероятности событий  $p$  для разных значений  $ns$ ,  $ng$  и  $nb$  при заданном  $nh$ , а

также как строились графики зависимости средних значений  $\langle ns \rangle$ ,  $\langle ng \rangle$  и  $\langle nb \rangle$  от  $nh$ .

Для подсчета вероятности события  $p$  был написан следующий код:

```
double probS[maxh+1][maxs+1]; // объявление массива
double probG[maxh+1][maxg+1];
double probB[maxh+1][maxb+1];

// считаем вероятности для каждого значения частицы в событиях с определенным h
for (int h = 0; h <= maxh; h++) { // проходим по значениям h
    cout<<"////////// " << endl;
    cout << "nh=" << h << " " << endl;
    cout<<" " << endl;
    for (int s = 0; s <= maxs; s++) { // проходим по значениям ns
        probS[h][s] = 0;
        float probability = 0; // счетчик для подсчета количества событий при s=0,1,2...
        float p=0; // переменная для подсчета вероятности
        int total_count = 0; // счетчик для подсчета количества всех событий
        for (int i = 0; i < ns.size(); i++) {
            if (nh[i] == h) { // если nh равно текущему значению h
                total_count++; // увеличиваем счетчик событий
                if (ns[i] == s) probability++; // если ns равно текущему значению s, увеличиваем счетчик вероятности
            }
        }
        if (total_count == 0) { // если для текущего значения nh нет событий,
            p=0; // то говорим что вероятность появления этого события = 0
        } else {
            p = probability / total_count; // считаем вероятности для каждого значения частицы
            probS[h][s] = p;
        }
        cout << "ns=" << s << " for probability p=" << p << " " << endl;
    }
}
```

**Рисунок 3.17 – Код для расчета вероятности каждого значения частицы для событий с определенным  $h$**

Этот код делает следующее:

а) – Он создает двумерный массив  $\text{probS}[\text{maxh}+1][\text{maxs}+1]$ , в котором будут храниться вероятности событий  $p$  для разных значений  $ns$  от  $nh$ .

б) – Он проходит по всем возможным значениям  $nh$  от 0 до  $\text{maxh}$  и выводит их на экран.

в) – Для каждого значения  $nh$  он проходит по всем возможным значениям  $ns$  от 0 до  $\text{maxs}$ .

г) – Создает переменную  $\text{probability}$ , в которой будет храниться количество событий, в которых  $ns$  равно текущему значению  $s$  при заданном  $nh$ .

д) – Он создает переменную  $\text{total\_count}$ , в которой будет храниться общее количество событий, в которых  $nh$  равно текущему значению  $h$ .

е) – Он проходит по всем элементам массива  $ns$ , который содержит данные о множественности ливневых частиц в каждом событии.

ж) – Проверяет, равно ли значение  $nh$  в текущем событии текущему значению  $h$ . Если да, то он увеличивает  $\text{total\_count}$  на единицу.

з) – Он проверяет, равно ли значение  $ns$  в текущем событии текущему значению  $s$ . Если да, то он увеличивает  $\text{probability}$  на единицу.

и) – Он вычисляет вероятность события  $p$  как отношение  $\text{probability}$  к  $\text{total\_count}$ . Если  $\text{total\_count}$  равно нулю, то он присваивает  $p$  нулевое значение.

к) – Он записывает значение  $p$  в соответствующий элемент массива  $\text{probS}$ .

л) – Он выводит на экран значение  $ns$ , вероятность  $p$  пример на рисунке.



```

root [0] .x fit2.cpp
//////////
nh=0:

ns=0 for probability p=0.466667;
ns=1 for probability p=0.333333;
ns=2 for probability p=0.166667;
ns=3 for probability p=0.0333333;

ng=0 for probability p=1;
ng=1 for probability p=0;
ng=2 for probability p=0;
ng=3 for probability p=0;

nb=0 for probability p=1;
nb=1 for probability p=0;
nb=2 for probability p=0;
nb=3 for probability p=0;
nb=4 for probability p=0;
nb=5 for probability p=0;
nb=6 for probability p=0;
nb=7 for probability p=0;
nb=8 for probability p=0;

//////////
nh=1:

ns=0 for probability p=0.565217;
ns=1 for probability p=0.347826;
ns=2 for probability p=0.0434783;
ns=3 for probability p=0.0434783;

ng=0 for probability p=0.652174;
ng=1 for probability p=0.347826;
ng=2 for probability p=0;
ng=3 for probability p=0;

nb=0 for probability p=0.347826;
nb=1 for probability p=0.652174;
nb=2 for probability p=0;
nb=3 for probability p=0;
nb=4 for probability p=0;
nb=5 for probability p=0;
nb=6 for probability p=0;
nb=7 for probability p=0;
nb=8 for probability p=0;

//////////
nh=2:

```

**Рисунок 3.18 – Вывод результатов в консоль**

Аналогичным образом я заполнила массивы probG и probB для g- и b-частиц.

Для вычисления средних значений  $\langle ns \rangle$ ,  $\langle ng \rangle$  и  $\langle nb \rangle$  в зависимости от nh я использовал следующий код:

```

//считаем среднее значение bgs для каждого h
double avgS[maxh+1];
for (int h = 0; h <= 8; h++) {
    double avg = 0;
    for (int s = 0; s <= maxs; s++){
        avg += s * probS[h][s];
    }
    avgS[h] = avg;
}
for (int h = 0; h < 9; h++) {
    cout << "h= " << h << " (<b>,<g>,<s>) : " << avgB[h] <<"; " << avgG[h] <<"; " << avgS[h] << endl;
}

```

**Рисунок 3.19 – Вычисление средних значений**

Этот код делает следующее:

- а) – Он создает одномерный массив avgS[maxh+1], в котором будут храниться средние значения  $\langle ns \rangle$  для разных значений nh.
- б) – Он проходит по всем возможным значениям nh от 0 до maxh.
- в) – Для каждого значения nh он создает переменную avg, в которой будет храниться среднее значение  $\langle ns \rangle$ .
- г) – Он проходит по всем возможным значениям ns от 0 до maxs и умножает их на соответствующие вероятности из массива probS. Он складывает все полученные произведения и записывает их в avg.
- д) – Он записывает значение avg в соответствующий элемент массива avgS.
- е) – Для каждого значения nh выводит среднее значение bgs частиц.
- ж) – Выводит результаты на консоль (рисунок 3.20).

Аналогичным образом я построил массивы avgG и avgB для g- и b-частиц.

```

h= 0 (<b>,<g>,<s>) :      0; 0; 0.766667
h= 1 (<b>,<g>,<s>) :      0.652174; 0.347826; 0.565217
h= 2 (<b>,<g>,<s>) :      1.5625; 0.4375; 0.375
h= 3 (<b>,<g>,<s>) :      2.13333; 0.866667; 0.8
h= 4 (<b>,<g>,<s>) :      3.16667; 0.833333; 1
h= 5 (<b>,<g>,<s>) :      4; 1; 0.833333
h= 6 (<b>,<g>,<s>) :      4.4; 1.6; 0
h= 7 (<b>,<g>,<s>) :      4.5; 2.5; 0.5
h= 8 (<b>,<g>,<s>) :      7; 1; 0

```

**Рисунок 3.20 – Вывод результатов в консоль**

Для построения графиков я использовал класс TGraph, который представляет собой график с двумя переменными. Я задал параметры графика, такие как название, оси, маркеры и цвета. Затем я заполнил график данными из массивов avgS, avgG и avgB, а также из массива ar, который содержит значения nh от 0 до maxh. Я также построил линейную аппроксимацию данных с помощью класса TF1, который представляет собой функцию одной переменной (рисунок 3.21). Я задал вид функции  $f(x)=a+kx$  и подобрал параметры a и k так, чтобы минимизировать квадратичную ошибку между функцией и данными. Я также вывел значения параметров a и k в консоль, где они соответствуют p0 и p1 смотри на рисунке 3.22.

```

g3 = new TGraph(maxh+1,ar,avgS); //ns
TF1 *fitFunc3 = new TF1("fitFunc3", "[0]+[1]*x",0,10);
fitFunc3->SetParameters(1.0,1.0);
fitFunc3->SetLineStyle(10);
fitFunc3->SetLineColor(kBlack);
g3->Fit(fitFunc3);
g3->SetMarkerStyle(25);
g3->Draw("PL same");

```

**Рисунок 3.21 – Фитирование функции**

```

Minimizer is Minuit / Migrad
Chi2                =      0.765368
Ndf                 =      7
Edm                 =      1.10916e-23
NCalls              =      30
p0                  =      0.803068    +/-    0.203238
p1                  =     -0.0663164    +/-    0.0426884

```

**Рисунок 3.22 – Вывод результатов в консоль**

Графики зависимости средних значений от nb, ng и ns были построены аналогичным способом.

## ЗАКЛЮЧЕНИЕ

Представленная работа была выполнена в рамках исследовательской программы эксперимента Беккерель на ускорительном комплексе Нуклотрон/NICA в ОИЯИ. Основными результатами выпускной квалификационной работы являются следующее:

а) – Освоена методика ядерных фотоэмульсий. Данная методика широко применяется в исследовательских экспериментах по регистрации треков заряженных частиц в широкой области энергий и ионизации в  $4\pi$ -геометрии. К основным преимуществам методики можно отнести уникальное пространственное разрешение, отсутствие порога чувствительности и относительная простота применения. Ручная обработка фильмовой информации, требующая больших людских ресурсов и времени, является потенциальным недостатком методики. Однако развитие современных автоматизированных микроскопов и применение алгоритмов цифровой обработки изображений нивелируют данную проблему.

б) – Проведенный анализ экспериментальных данных выполнен в рамках исследовательских задач, решаемых в международной коллаборации эксперимента Беккерель в Лаборатории физики высоких энергий им. академиков В.И. Векслера и А.М. Балдина Объединенного института ядерных исследований г. Дубна, Россия.

в) – В качестве экспериментального материала были использованы образцы слоев ядерной фотоэмульсии, продольно облученных в пучках релятивистских ядер  ${}^9\text{Be}$  и  ${}^{10}\text{B}$  с кинетической энергией 1 ГэВ на нуклон. Выполнен просмотр образцов ядерной фотоэмульсии, результатом которого стало дополнение существующей статистики событий с множественным образованием вторичных фрагментов He и H налетающих ядер.

г) – Изучены корреляционные зависимости множественности  $b$ ,  $g$  и  $s$  - частиц во фрагментации ядер-мишени из состава ядерной фотоэмульсии под действием релятивистских ядер  ${}^{10}\text{B}$  и  ${}^9\text{Be}$ . Дан сравнительный анализ по средней множественности частиц в зависимости от типа ядра-мишени. Был установлен эффект насыщения в распределении по множественности  $b$ -частиц в событиях  ${}^9\text{Be}+\text{Em}$ . Для  $g$ -частиц прослеживается монотонный рост с числом  $h$ -частиц в обоих наборах данных, в отличие от  $s$ -частиц, имеющих тенденцию к уменьшению.

## СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Пауэл С. Исследование элементарных частиц фотографическим методом: пер. с англ. / Пауэл С., Фаулер П., Перкинс Д.Б - Москва: Издательство иностранной литературы, 1962. - 653 с.
2. V. Bradnova Beryllium (Boron) Clustering Quest in Relativistic Multifragmentation / V. Bradnova // ЯФ. –2003. – 66, 3, 1694-1698.
3. Artemenkov, D.A. Unstable states in dissociation of relativistic nuclei / Artemenkov, D.A., Bradnova, V., Chernyavsky, M.M. // Eur. Phys. J. – 2020 –A 56, 250.
4. Tilley D. R. Energy levels of light nuclei  $A= 8, 9, 10$  / Tilley D. R. //Nuclear Physics A. – 2004. – Т. 745. – №. 3-4. – С. 155-362.
5. Adamovich M. I. Interactions of relativistic  ${}^6\text{Li}$  nuclei with photoemulsion nuclei / M. I. Adamovich // Phys. At. Nucl. – 1999 – 62, 1378; arXiv: 1109.6422 [nucl-ex]
6. Zaitsev, A.A. Dissociation of relativistic  ${}^{10}\text{B}$  nuclei in nuclear track emulsion / Zaitsev, A.A., Artemenkov, D.A., Bradnova, V. // Phys. Part. Nuclei. – 2017 – 48, 960–963.
7. Периферические взаимодействия релятивистских ядер  ${}^{14}\text{N}$  с ядрами фотоэмульсии / Т. В. Щедрина [и др.] //Ядерная физика. – 2007. – Т.70.
8. Поиск распадов ядра  ${}^9\text{B}$  и состояния Хойла в диссоциации ядер  ${}^{14}\text{N}$  / Е. Мицова [и др.]// ЭЧАЯ – 2022 – Т.53, с.416-424.
9. Rokujo H. Latest nuclear emulsion technology / H. Rokujo // EPJ Web of Conferences. – 2017 – 145, 19020.
10. Перфилов Н. А. Ядерные реакции под действием частиц высоких энергий / Н. А. Перфилов, О. В. Ложкин и В. И. Остроумов. – АН СССР. Ленинградское отделение, 1962.
11. H. L. Bradt The Heavy Nuclei of the Primary Cosmic Radiation/ H. L. Bradt, B. Peters // *Physical Review Journal Archive*. – 1950 – т. 77, р. 54.
12. Артеменков Д.А. Исследование фрагментации ядер  ${}^9\text{Be}$  на альфа – частичные пары в ядерной фотоэмульсии при энергии 1.2 А ГэВ/ Артеменков Д.А. // Дубна – 2007 г.
13. Зайцев А. А. Исследование диссоциации релятивистских ядер  ${}^{10}\text{B}$ ,  ${}^{11}\text{C}$  и  ${}^{12}\text{C}$  методом ядерной фотоэмульсии / Зайцев А. А. // Дубна – 2019.
14. Brun, Rene, and Fons Rademakers. ROOT—An object oriented data analysis framework. Nuclear instruments and methods in physics research section A: accelerators, spectrometers, detectors and associated equipment 389.1-2 (1997): 81-86.

## ПРИЛОЖЕНИЯ

### ПРИЛОЖЕНИЕ А

#### Матрица распределений по множественности для канала ${}^9\text{Be} \rightarrow 2\text{He}$

Таблица А.1

	Матрица распределений по множественности ns – nh для канала ${}^9\text{Be} \rightarrow 2\text{He}$											
$\begin{smallmatrix} nh \\ ns \end{smallmatrix}$	0	1	2	3	4	5	6	7	8	9	10	$\Sigma$
0	198	90	28	16	22	7	5	3	1	4	0	374
1	36	16	20	13	11	4	1	0	0	0	1	102
2	4	6	6	2	1	0	0	0	0	0	0	19
3	1	2	0	0	1	0	0	0	0	0	0	4
4	0	1	0	0	0	0	0	0	0	0	0	1
$\Sigma$	239	115	54	31	35	11	6	3	1	4	1	500

Таблица А.2

	Матрица распределений по множественности ng – nb для канала ${}^9\text{Be} \rightarrow 2\text{He}$										
$\begin{smallmatrix} nb \\ ng \end{smallmatrix}$	0	1	2	3	4	5	6	7	8	9	$\Sigma$
0	239	66	30	12	9	2	1	0	0	0	359
1	49	18	12	11	4	1	0	0	0	1	96
2	6	5	10	4	0	1	0	2	0	0	28
3	2	3	1	1	1	0	1	0	0	0	9
4	2	0	2	1	1	0	0	0	0	0	6

5	0	1	0	0	0	0	0	0	0	0	1
6	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0
8	0	1	0	0	0	0	0	0	0	0	1
$\Sigma$	298	94	55	29	15	4	2	2	0	1	500

## ПРИЛОЖЕНИЕ Б

### Матрица распределений по множественности для канала $^{10}\text{B} \rightarrow 2\text{He}$

Таблица Б.1

	Матрица распределений по множественности ns – nh для канала $^{10}\text{B} \rightarrow 2\text{He}$									
$\begin{smallmatrix} nh \\ ns \end{smallmatrix}$	0	1	2	3	4	5	6	7	8	$\Sigma$
0	14	13	11	6	3	3	5	2	2	59
1	10	8	4	6	0	1	0	2	0	31
2	5	1	1	3	3	2	0	0	0	15
3	1	1	0	0	0	0	0	0	0	2
$\Sigma$	30	23	16	15	6	6	5	4	2	107

Таблица Б.2

	Матрица распределений по множественности ng – nb для канала $^{10}\text{B} \rightarrow 2\text{He}$									
$\begin{smallmatrix} nb \\ ng \end{smallmatrix}$	0	1	2	3	4	5	6	7	8	$\Sigma$
0	30	15	10	5	2	1	0	0	1	64
1	8	5	7	3	4	2	0	0	0	29
2	1	3	1	1	3	2	1	0	0	12
3	0	0	0	0	2	0	0	0	0	2
$\Sigma$	39	23	18	9	11	5	1	0	1	107

## ПРИЛОЖЕНИЕ В

**Матрица распределений по множественности для канала  $^{10}\text{B} \rightarrow 2\text{He} + \text{H}$**   
Таблица В.1

	Матрица распределений по множественности ns – nh для канала $^{10}\text{B} \rightarrow 2\text{He} + \text{H}$								
$\begin{smallmatrix} nh \\ ns \end{smallmatrix}$	0	1	2	3	4	5	6	7	$\Sigma$
0	36	45	13	12	16	5	2	1	130
1	17	8	10	4	7	1	0	1	48
2	1	0	3	2	1	1	0	0	8
3	1	2	1	0	1	0	0	0	5
4	1	0	1	0	0	0	0	0	2
$\Sigma$	56	55	28	18	25	7	2	2	193

Таблица В.2

	Матрица распределений по множественности ng – nb для канала $^{10}\text{B} \rightarrow 2\text{He} + \text{H}$								
$\begin{smallmatrix} nb \\ ng \end{smallmatrix}$	0	1	2	3	4	5	6	7	$\Sigma$
0	56	39	20	13	10	3	0	1	142
1	16	7	4	11	3	2	1	0	44
2	1	1	3	0	0	0	0	0	5
3	0	1	1	0	0	0	0	0	2
$\Sigma$	73	48	28	24	13	5	1	1	193



## ПРИЛОЖЕНИЕ Г

**Программный код для построения нормированного распределения по множественности  $nh$ ,  $nb$ ,  $ng$ ,  $ns$  ( $^9\text{Be}$ ), где  $nh \geq 0$**

```
#include <iostream>
#include <fstream>
#include <iomanip>
#include <stdlib.h>
using namespace std;
int bgs0()
{
    ifstream F;
    F.open("open.dat",ios::in);
    Double_t ntrek,type;
    Int_t name, nb, ng, ns, nf, na, n6;
    Float_t openang, erropenang;
    gROOT->SetStyle("Pub");
    TH1F* h1=new TH1F("nb", "Histogram title; nb; Counts", 8, 0, 8);
    TH1F* h2=new TH1F("ng", "Histogram title; ng; Counts", 8, 0, 8);
    TH1F* h3=new TH1F("ns", "Histogram title; ns; Counts", 8, 0, 8);
    TH1F *h4 = new TH1F("nh", "Histogram title; nh; Counts",10,0,10);

    auto hs = new THStack("nh", "Histogram title; nh; Counts");

    if (F)
    {
        while(!F.eof())
        {
            F >>ntrek>>type;
            F.ignore(1024,'\n');
            F >>name>>nb>>ng>>ns;
            F.ignore(1024,'\n');
            F >>openang >>erropenang;
            F.ignore(1024,'\n');
            if (!F.good()) break;
            h1->Fill(nb);
            h2->Fill(ng);
            h3->Fill(ns);
            if (nb+ng>=0)
            {
                h4->Fill(nb+ng);
            }
        }
        F.close();
    }
    else cout<<"No file"<<endl;
```

```

h1->SetLineColor(kBlack);

h2->SetLineColor(kBlack);

h3->SetLineColor(kBlack);

h4->SetLineColor(kBlack);

auto cst = new TCanvas("bgs Be9"," Particle multiplicity distribution nb, ng,
ns (Be9)",10,10,900,900);
cst->Divide(2,2);

cst->cd(1);
    Double_t factor = 1.;
    h4->Scale(factor/h4->Integral());
h4->Draw();

cst->cd(2);
    h1->Scale(factor/h1->Integral());
h1->Draw();

cst->cd(3);
    h2->Scale(factor/h2->Integral());
h2->Draw();

cst->cd(4);
    h3->Scale(factor/h3->Integral());
h3->Draw();

gROOT->GetListOfCanvases()->Draw();

return 0;

}

```

## ПРИЛОЖЕНИЕ Д

Программный код для построения нормированного распределения по множественности  $nh$ ,  $nb$ ,  $ng$ ,  $ns$  для канала  $^{10}B \rightarrow 2He + H$ , где  $nh \geq 0$

```
#include <iostream>
#include <fstream>
#include <iomanip>
#include <stdlib.h>
using namespace std;
int bgs0()
{
    ifstream F;
    F.open("B-01-18.dat",ios::in);
    Int_t name, nb, ng, ns, n4, n3, n2, n1;
    Float_t openang, erropenang;
    gROOT->SetStyle("Pub");
    TH1F* h1=new TH1F("nb", "Histogram title; nb; Counts", 8, 0, 8);
    TH1F* h2=new TH1F("ng", "Histogram title; ng; Counts", 8, 0, 8);
    TH1F* h3=new TH1F("ns", "Histogram title; ns; Counts", 8, 0, 8);
    TH1F *h4 = new TH1F("nh", "Histogram title; nh; Counts",12,0,12);

    if (F)
    {
        while(!F.eof())
        {
            F >>name>>nb>>ng>>ns>>n4>>n3>>n2>>n1;
            F.ignore(1024,'\n');
            if (!F.good()) break;
            if(n2==2 && n1==1){
                h1->Fill(nb);
                h2->Fill(ng);
                h3->Fill(ns);
                if (nb+ng>=0)
                {
                    h4->Fill(nb+ng);
                }
            }
        }
        F.close();
    }else cout<<"No file"<<endl;

    h1->SetLineColor(kBlack);

    h2->SetLineColor(kBlack);

    h3->SetLineColor(kBlack);
```

```

h4->SetLineColor(kBlack);

auto cst = new TCanvas("hs"," Particle multiplicity distribution nh, nb, ng,
ns (B10) ",10,10,900,900);
cst->Divide(2,2);

cst->cd(1);
    Double_t factor = 1.;
    h4->Scale(factor/h4->Integral());
h4->Draw();

cst->cd(2);
    h1->Scale(factor/h1->Integral());
h1->Draw();

cst->cd(3);
    h2->Scale(factor/h2->Integral());
h2->Draw();

cst->cd(4);
    h3->Scale(factor/h3->Integral());
h3->Draw();

gROOT->GetListOfCanvases()->Draw();
return 0;
}

```

## ПРИЛОЖЕНИЕ Е

Программный код для построения нормированного распределения по множественности  $nh$ ,  $nb$ ,  $ng$ ,  $ns$  для канала  $^{10}B \rightarrow 2He$ , где  $nh \geq 0$

```
#include <iostream>
#include <fstream>
#include <iomanip>
#include <stdlib.h>
using namespace std;
int bgs0()
{
    ifstream F;
    F.open("open.dat",ios::in);
    Double_t ntrek,type;
    Int_t name, nb, ng, ns, nf, na, n6;
    Float_t openang, erropenang;
    gROOT->SetStyle("Pub");
    TH1F* h1=new TH1F("nb", "Histogram title; nb; Counts", 8, 0, 8);
    TH1F* h2=new TH1F("ng", "Histogram title; ng; Counts", 8, 0, 8);
    TH1F* h3=new TH1F("ns", "Histogram title; ns; Counts", 8, 0, 8);
    TH1F *h4 = new TH1F("nh", "Histogram title; nh; Counts",10,0,10);

    auto hs = new THStack("nh", "Histogram title; nh; Counts");

    if (F)
    {
        while(!F.eof())
        {
            F >>ntrek>>type;
            F.ignore(1024,'\n');
            F >>name>>nb>>ng>>ns;
            F.ignore(1024,'\n');
            F >>openang >>erropenang;
            F.ignore(1024,'\n');
            if (!F.good()) break;
            h1->Fill(nb);
            h2->Fill(ng);
            h3->Fill(ns);
            if (nb+ng>=0)
            {
                h4->Fill(nb+ng);
            }
        }
        F.close();
    }else cout<<"No file"<<endl;
```

```

h1->SetLineColor(kBlack);

h2->SetLineColor(kBlack);

h3->SetLineColor(kBlack);

h4->SetLineColor(kBlack);

auto cst = new TCanvas("bgs Be9"," Particle multiplicity distribution nb, ng,
ns (Be9)",10,10,900,900);
cst->Divide(2,2);

cst->cd(1);
    Double_t factor = 1.;
    h4->Scale(factor/h4->Integral());
h4->Draw();

cst->cd(2);
    h1->Scale(factor/h1->Integral());
h1->Draw();

cst->cd(3);
    h2->Scale(factor/h2->Integral());
h2->Draw();

cst->cd(4);
    h3->Scale(factor/h3->Integral());
h3->Draw();

gROOT->GetListOfCanvases()->Draw();

return 0;
}

```

## ПРИЛОЖЕНИЕ Ж

Программный код для построения корреляции множественности от  $nh$   
для канала  $^{10}B \rightarrow 2He$

```
#include <iostream>
#include <fstream>
#include <vector>

using namespace std;

int fit2() {
    ifstream file("B-01-18.dat");           // открываем
    файл для чтения
    vector<int> ns, nb, ng, nh;             // объявляем
    векторы для хранения значений
    int n2, n1;                             //
    переменные для считывания из файла
    int id, temp_nb, temp_ng, temp_ns, temp_n2, temp_n1, temp_n3,
    temp_n4; // временные переменные для считывания из файла

    if (file) {                             // если файл
    успешно открыт
        while (file >> id >> temp_nb >> temp_ng >> temp_ns >> temp_n4 >>
        temp_n3 >> temp_n2 >> temp_n1)
        {
            // считываем данные из файла
            if (temp_n2 == 2 && temp_n1 == 0) { // если условие
            выполняется
                nb.push_back(temp_nb);         // добавляем
                значения в векторы
                ng.push_back(temp_ng);
                ns.push_back(temp_ns);
                nh.push_back(temp_nb + temp_ng);
            }
        }
        file.close();                         //
    закрываем файл
    }

    //////////////////////////////////////
    //вычисление среднего значения через вероятность
    int maxb = *max_element(nb.begin(), nb.end()); // находим
    максимальные значения
    int maxg = *max_element(ng.begin(), ng.end());
    int maxs = *max_element(ns.begin(), ns.end());
    int maxh = *max_element(nh.begin(), nh.end());
```

```

double probS[maxh+1][maxs+1]; // объявление массива
double probG[maxh+1][maxg+1];
double probB[maxh+1][maxb+1];

//считаем вероятности для каждого значения частицы в событиях с
определенным h
    for (int h = 0; h <= maxh; h++) { // проходим
по значениям h
        cout<<"/////////"<<endl;
        cout << "nh=" << h << ":" << endl;
        cout<<" "<<endl;
        for (int s = 0; s <= maxs; s++) { // проходим по
значениям ns
            probS[h][s] = 0;
            float probability = 0; //
счетчик для подсчета количества событий при s=0,1,2...
            float p=0;
            // переменная для подсчета вероятности
            int total_count = 0; // счетчик
для подсчета количества всех событий

            for (int i = 0; i < ns.size(); i++) {

                if (nh[i] == h) { // если nh
равно текущему значению h
                    total_count++;
                    // увеличиваем счетчик событий
                    if (ns[i] == s) probability++; // если ns
равно текущему значению s, увеличиваем счетчик вероятности
                }
            }

            if(total_count==0){
                //если для текущего значения nh нет событий,
                p=0;
                //то говорим что вероятность появления этого события =0
            }else
                p=probability / total_count; //считаем
вероятности для каждого значения частицы
            probS[h][s] = p;

            // выводим результаты
            cout << "ns=" << s << " for probability p=" << p << ";" << endl;
        }
        cout<<" "<<endl;

```



```

        for (int g = 0; g <= maxg; g++) { // проходим по
значениям ng
            probG[h][g] = 0;
            float probability = 0; //
переменная для подсчета вероятности
            float p=0; // переменная для
подсчета вероятности
            int total_count = 0; //
счетчик для подсчета количества событий
            for (int i = 0; i < ng.size(); i++) {

                if (nh[i] == h) { // если nh
равно текущему значению h
                    total_count++;
                // увеличиваем счетчик событий
                if (ng[i] == g) probability++; // если ns
равно s, увеличиваем счетчик вероятности
            }
        }
        if(total_count==0){
            p=0;
        }else
            p=probability / total_count;
        probG[h][g] = p;
        cout << "ng=" << g << " for probability p=" << p << ";" << endl; //
выводим результаты
    }

    cout<<" "<<endl;
    for (int b = 0; b <= maxb; b++) { // проходим по
значениям nb
        probB[h][b] = 0;
        float probability = 0; //
переменная для подсчета вероятности
        float p=0;
        // переменная для подсчета вероятности
        int total_count = 0; // счетчик
для подсчета количества событий
        for (int i = 0; i < nb.size(); i++) {

            if (nh[i] == h) { // если nh
равно текущему значению h
                total_count++;
            // увеличиваем счетчик событий

```

```

        if (nb[i] == b) probability++;           // если ns
        равно s=0, увеличиваем счетчик вероятности
    }
}
    if(total_count==0){
        p=0;
    }else
        p=probability / total_count;
    probB[h][b] = p;
    cout << "nb=" << b << " for probability p=" << p << ";" << endl; //
ВЫВОДИМ РЕЗУЛЬТАТЫ
}
    cout<<" "<<endl;
}

double avgB[9];
for (int h = 0; h <= 8; h++) {
    double avg = 0;
    for (int b = 0; b <= maxb; b++){
        avg += b * probB[h][b];
    }
    avgB[h] = avg;
}

double avgG[9];
for (int h = 0; h <= 8; h++) {
    double avg = 0;
    for (int g = 0; g <= maxg; g++){
        avg += g * probG[h][g];
    }
    avgG[h] = avg;
}

//считаем среднее значение bgs для каждого h
double avgS[maxh+1];
for (int h = 0; h <= 8; h++) {
    double avg = 0;
    for (int s = 0; s <= maxs; s++){
        avg += s * probS[h][s];
    }
    avgS[h] = avg;
}
for (int h = 0; h < 9; h++) {
    cout << "h= " << h << " (<b>,<g>,<s>) : " << avgB[h] <<"; " <<
avgG[h] <<"; " << avgS[h] << endl;

```

```

    }

////////////////////////////////////
/////fitGraph
TCanvas *C;
TGraph *g1, *g2, *g3, *gr1, *gr2, *gr3;

C = new TCanvas("vC1","square",200,10,500,700);

////////////////////////////////////
//lineFunc
Double_t ar[maxh+1];
for (int s = 0; s <= maxh; s++){
    ar[s]=s;
}

g1 = new TGraph(maxh+1,ar,avgB);//nb
g1->SetTitle("Correlation between <nb> <ng> <ns> from Nh");
g1->GetXaxis()->SetTitle("Nh (10B->2He)");
TF1 *fitFunc = new TF1("fitFunc", "[0]+[1]*x",0,10);
fitFunc->SetParameters(1.0,1.0);
fitFunc->SetLineStyle(1);
fitFunc->SetLineColor(kBlack);
g1->Fit(fitFunc);
g1->SetMarkerStyle(20);
g1->Draw("APL");
g1->GetXaxis()->SetRangeUser(0, 10);
g1->GetYaxis()->SetRangeUser(0, 20);

g2 = new TGraph(maxh+1,ar,avgG);//ng
TF1 *fitFunc2 = new TF1("fitFunc2", "[0]+[1]*x",0,10);
fitFunc2->SetParameters(1.0,1.0);
fitFunc2->SetLineStyle(9);
fitFunc2->SetLineColor(kBlack);
g2->Fit(fitFunc2);
g2->SetMarkerStyle(23);
g2->Draw("PL same");

g3 = new TGraph(maxh+1,ar,avgS);//ns
TF1 *fitFunc3 = new TF1("fitFunc3", "[0]+[1]*x",0,10);
fitFunc3->SetParameters(1.0,1.0);
fitFunc3->SetLineStyle(10);
fitFunc3->SetLineColor(kBlack);
g3->Fit(fitFunc3);
g3->SetMarkerStyle(25);

```

```

g3->Draw("PL same");

TLegend *legend = new TLegend(0.7, 0.7, 0.9, 0.9);
  legend->AddEntry(g1, "- <nb> from Nh", "p");
  legend->AddEntry(g2, "- <ng> from Nh", "p");
legend->AddEntry(g3, "- <ns> from Nh", "p");
  legend->Draw();
TLegend *legend2 = new TLegend(0.7, 0.7, 0.9, 0.9);
  legend2->AddEntry(fitFunc, " - fit function for <nb> from Nh", "l");
  legend2->AddEntry(fitFunc2, " - fit function for <ng> from Nh", "l");
legend2->AddEntry(fitFunc3, " - fit function for <ns> from Nh", "l");
  legend2->Draw();

  return 0;
}

```

## ПРИЛОЖЕНИЕ И

### Программный код для построения корреляции множественности от nb для канала $^{10}\text{B} \rightarrow 2\text{He}$

```
#include <iostream>
#include <fstream>
#include <vector>

using namespace std;

int fitNb() {
    ifstream file("B-01-18.dat");           // открываем
    файл для чтения
    vector<int> ns, nb, ng, nh;             // объявляем
    векторы для хранения значений
    int n2, n1;                             //
    переменные для считывания из файла
    int id, temp_nb, temp_ng, temp_ns, temp_n2, temp_n1, temp_n3,
    temp_n4; // временные переменные для считывания из файла

    if (file) {                             // если файл
    успешно открыт
        while (file >> id >> temp_nb >> temp_ng >> temp_ns >> temp_n4 >>
        temp_n3 >> temp_n2 >> temp_n1)
        {
            // считываем данные из файла
            if (temp_n2 == 2 && temp_n1 == 0) { // если условие
            выполняется
                nb.push_back(temp_nb);         // добавляем
                значения в векторы
                ng.push_back(temp_ng);
                ns.push_back(temp_ns);
                nh.push_back(temp_nb + temp_ng);
            }
        }
        file.close();                         //
    закрываем файл
    }

    //////////////////////////////////////
    //вычисление среднего значения через вероятность
    int maxb = *max_element(nb.begin(), nb.end()); // находим
    максимальные значения
    int maxg = *max_element(ng.begin(), ng.end());
    int maxs = *max_element(ns.begin(), ns.end());
    int maxh = *max_element(nh.begin(), nh.end());
```

```

double probS[maxb+1][maxs+1]; // объявление массива
double probG[maxb+1][maxg+1];
double probH[maxb+1][maxh+1];

for (int h = 0; h <= maxb; h++) {           // проходим по значениям h
    cout<<"//////////////////// " << endl;
    cout << "nb=" << h << ":" << endl;
    cout<<" " << endl;
    for (int b = 0; b <= maxh; b++) {       // проходим по
значениям nb
        probH[h][b] = 0;
        float probability = 0;             //
переменная для подсчета вероятности
        float p=0;                         // переменная для
подсчета вероятности
        int total_count = 0;               // счетчик
для подсчета количества событий
        for (int i = 0; i < nh.size(); i++) {

            if (nb[i] == h) {               // если nh
равно текущему значению h
                total_count++;
                // увеличиваем счетчик событий
                if (nh[i] == b) probability++; // если ns
равно s=0, увеличиваем счетчик вероятности
            }
        }
        if(total_count==0){
            p=0;
        }else
            p=probability / total_count;
        probH[h][b] = p;
        cout << "nh=" << b << " for probability p=" << p << ";" << endl; //
выводим результаты
    }

    cout<<" " << endl;
    for (int g = 0; g <= maxg; g++) {       // проходим по
значениям ng
        probG[h][g] = 0;
        float probability = 0;             //
переменная для подсчета вероятности
        float p=0;                         // переменная для
подсчета вероятности

```

```

        int total_count = 0; //
        счетчик для подсчета количества событий
        for (int i = 0; i < ng.size(); i++) {

            if (nb[i] == h) { // если nh
                равно текущему значению h
                    total_count++;
                // увеличиваем счетчик событий
                if (ng[i] == g) probability++; // если ns
                равно s, увеличиваем счетчик вероятности
            }
        }

        if(total_count==0){
            p=0;
        }else
            p=probability / total_count;
        probG[h][g] = p;
        cout << "ng=" << g << " for probability p=" << p << "; " << "Ns =
"<< probability << ";    all = "<< total_count << endl; // выводим результаты
    }
    cout<<" "<< endl;
    for (int s = 0; s <= maxs; s++) { //      проходим      по
        значениям ns
            probS[h][s] = 0;
            float probability = 0; //
            переменная для подсчета вероятности
            float p=0;
            int total_count = 0;
            int count = 0;
            // счетчик для подсчета количества событий
            for (int i = 0; i < ns.size(); i++) {

                if (nb[i] == h) { // если nh
                    равно текущему значению h
                        total_count++;
                    // увеличиваем счетчик событий
                    if (ns[i] == s) probability++; // если ns
                    равно s=0, увеличиваем счетчик вероятности
                }
            }

            if(total_count==0){
                p=0;
            }else
                p=probability / total_count;
            probS[h][s] = p;

```

```

        //cout << prob[h][s] << " ";
        cout << "ns=" << s << " for probability p=" << p << ";" << endl; //
ВЫВОДИМ РЕЗУЛЬТАТЫ
    }
}
cout<<" "<< endl;
double avgH[maxb+1];
for (int h = 0; h <= maxb; h++) {
    double avg = 0;
    for (int b = 0; b <= maxh; b++){
        avg += b * probH[h][b];
    }
    avgH[h] = avg;
}
double avgG[maxb+1];
for (int h = 0; h <= maxb; h++) {
    double avg = 0;
    for (int s = 0; s <= maxg; s++){
        avg += s * probG[h][s];
    }
    avgG[h] = avg;
}
double avgS[maxb+1];
for (int h = 0; h <= maxb; h++) {
    double avg = 0;
    for (int s = 0; s <= maxs; s++){
        avg += s * probS[h][s];
    }
    avgS[h] = avg;
}

for (int h = 0; h <=maxb; h++) {
    cout << "b= " << h << " (<h>,<g>,<s>) : " << avgH[h] <<"; " <<
avgG[h] <<"; " << avgS[h] << endl;
}

TCanvas *C;
TGraph *g1, *g2, *g3, *gr1, *gr2, *gr3;

C = new TCanvas("vC1","square",200,10,500,700);

////////////////////////////////////
//lineFunc
Double_t ar[maxb+1];
for (int s = 0; s <= maxb; s++){

```



```

        ar[s]=s;
    }
    ///*
    g1 = new TGraph(maxb+1,ar,avgH);
    g1->SetTitle("Correlation between <nh> <ng> <ns> from Nb");
    g1->GetXaxis()->SetTitle("Nb (10B->2He)");
    TF1 *fitFunc = new TF1("fitFunc", "[0]+[1]*x",0,10);
    fitFunc->SetParameters(1.0,1.0);
    fitFunc->SetLineStyle(1);
    fitFunc->SetLineColor(kBlack);
    g1->Fit(fitFunc);
    g1->SetMarkerStyle(20);
    g1->Draw("APL");
    g1->GetXaxis()->SetRangeUser(0, 10);
    g1->GetYaxis()->SetRangeUser(0, 20);

    //C->cd(2);
    g2 = new TGraph(maxb+1,ar,avgG);
    TF1 *fitFunc2 = new TF1("fitFunc2", "[0]+[1]*x",0,10);
    fitFunc2->SetParameters(1.0,1.0);
    fitFunc2->SetLineStyle(9);
    fitFunc2->SetLineColor(kBlack);
    g2->Fit(fitFunc2);
    g2->SetMarkerStyle(23);
    g2->Draw("PL same");
    //g2->GetXaxis()->SetRangeUser(0, 10);
    //g2->GetYaxis()->SetRangeUser(0, 20);

    //C->cd(3);
    g3 = new TGraph(maxb+1,ar,avgS);
    TF1 *fitFunc3 = new TF1("fitFunc3", "[0]+[1]*x",0,10);
    fitFunc3->SetParameters(1.0,1.0);
    fitFunc3->SetLineStyle(10);
    fitFunc3->SetLineColor(kBlack);
    g3->Fit(fitFunc3);
    g3->SetMarkerStyle(25);
    //g3->GetXaxis()->SetRangeUser(0, 10);
    //g3->GetYaxis()->SetRangeUser(0, 20);
    g3->Draw("PL same");

    TLegend *legend = new TLegend(0.7, 0.7, 0.9, 0.9);
    legend->AddEntry(g1, "- <nh> ", "p");
    legend->AddEntry(g2, "- <ng> ", "p");
    legend->AddEntry(g3, "- <ns> ", "p");
    legend->Draw();

```

```
TLegend *legend2 = new TLegend(0.7, 0.7, 0.9, 0.9);
    legend2->AddEntry(fitFunc, " - fit function for <nh> from Nb", "l");
    legend2->AddEntry(fitFunc2, " - fit function for <ng> from Nb", "l");
    legend2->AddEntry(fitFunc3, " - fit function for <ns> from Nb", "l");
    legend2->Draw();

    return 0;
}
```

## ПРИЛОЖЕНИЕ К

### Программный код для построения корреляции множественности от $ng$ для канала $^{10}B \rightarrow 2He$

```
#include <iostream>
#include <fstream>
#include <vector>
#include "Riostream.h"
#include "math.h"
#include "TImage.h"
#include "TCanvas.h"
#include "TArrayD.h"
#include "TROOT.h"
#include "TColor.h"
#include "TAttImage.h"
#include "TPluginManager.h"

using namespace std;

int fitNg() {
    ifstream file("B-01-18.dat");           // открываем
    файл для чтения
    vector<int> ns, nb, ng, nh;             // объявляем
    векторы для хранения значений
    int n2, n1;                             //
    переменные для считывания из файла
    int id, temp_nb, temp_ng, temp_ns, temp_n2, temp_n1, temp_n3,
    temp_n4; // временные переменные для считывания из файла

    if (file) {                             // если файл
    успешно открыт
        while (file >> id >> temp_nb >> temp_ng >> temp_ns >> temp_n4 >>
        temp_n3 >> temp_n2 >> temp_n1)
        {
            // считываем данные из файла
            if (temp_n2 == 2 && temp_n1 == 0) { // если условие
            выполняется
                nb.push_back(temp_nb);         // добавляем
                значения в векторы
                ng.push_back(temp_ng);
                ns.push_back(temp_ns);
                nh.push_back(temp_nb + temp_ng);
            }
        }
        file.close();                       //
    закрываем файл
}
```

```

    }
    //////////////////////////////////////
    //вычисление среднего значения через вероятность
    int maxb = *max_element(nb.begin(), nb.end());    //      находим
максимальные значения
    int maxg = *max_element(ng.begin(), ng.end());
    int maxs = *max_element(ns.begin(), ns.end());
    int maxh = *max_element(nh.begin(), nh.end());

    double probS[maxg+1][maxs+1]; // объявление массива
    double probH[maxg+1][maxh+1];
    double probB[maxg+1][maxb+1];

    for (int h = 0; h <= maxg; h++)
    {
        cout<<"//////////////// " << endl;
        cout << "ng=" << h << ":" << endl;
        cout<<" " << endl;

        for (int s = 0; s <= maxs; s++) {                //      проходим      по
значениям ns
            probS[h][s] = 0;
            float probability = 0;                        //
переменная для подсчета вероятности
            float p=0;
            int total_count = 0;
            int count = 0;
            // счетчик для подсчета количества событий
            for (int i = 0; i < ns.size(); i++) {

                if (ng[i] == h) {                          //      если      nh
равно текущему значению h
                    total_count++;
                    // увеличиваем счетчик событий
                    if (ns[i] == s) probability++;        //      если      ns
равно s=0, увеличиваем счетчик вероятности
                }
            }
            if(total_count==0){
                p=0;
            }else
            p=probability / total_count;
            probS[h][s] = p;
            //cout << prob[h][s] << " ";
        }
    }

```

```

        cout << "ns=" << s << " for probability p=" << p << ";" << endl; //
выводим результаты
    }
    cout<<" "<<endl;

    for (int g = 0; g <= maxh; g++) { // проходим по
значениям ng
        probH[h][g] = 0;
        float probability = 0; //
переменная для подсчета вероятности
        float p=0; // переменная для
подсчета вероятности
        int total_count = 0; //
счетчик для подсчета количества событий
        for (int i = 0; i < nh.size(); i++) {

            if (ng[i] == h) { // если nh
равно текущему значению h
                total_count++;
                // увеличиваем счетчик событий
                if (nh[i] == g) probability++; // если ns
равно s, увеличиваем счетчик вероятности
            }
        }
        if(total_count==0){
            p=0;
        }else
            p=probability / total_count;
        probH[h][g] = p;
        cout << "nh=" << g << " for probability p=" << p << ";" << endl; //
выводим результаты
    }
    cout<<" "<<endl;

    for (int b = 0; b <= maxb; b++) { // проходим по
значениям nb
        probB[h][b] = 0;
        float probability = 0; //
переменная для подсчета вероятности
        float p=0; // переменная для
подсчета вероятности
        int total_count = 0; // счетчик
для подсчета количества событий
        for (int i = 0; i < nb.size(); i++) {

```

```

        if (ng[i] == h) { // если nh
равно текущему значению h
            total_count++;
            // увеличиваем счетчик событий
            if (nb[i] == b) probability++; // если ns
равно s=0, увеличиваем счетчик вероятности
        }
    }
    if(total_count==0){
        p=0;
    }else
        p=probability / total_count;
    probB[h][b] = p;
    cout << "nb=" << b << " for probability p=" << p << "; Ns = " <<
probability << "; all = " << total_count << endl; // выводим результаты
}
    cout<<" " <<endl;
}
cout<<" " <<endl;

double avgH[maxg+1];
for (int h = 0; h <= maxg; h++) {
    double avg = 0;
    for (int g = 0; g <= maxh; g++){
        avg += g * probH[h][g];
    }
    avgH[h] = avg;
}
double avgS[maxg+1];
for (int h = 0; h <= maxg; h++) {
    double avg = 0;
    for (int s = 0; s <= maxs; s++){
        avg += s * probS[h][s];
    }
    avgS[h] = avg;
}
double avgB[maxg+1];
for (int h = 0; h <= maxg; h++) {
    double avg = 0;
    for (int s = 0; s <= maxb; s++){
        avg += s * probB[h][s];
    }
    avgB[h] = avg;
}

```

```

        for (int h = 0; h <= maxg; h++) {
            cout << "g= " << h << " (<h>,<b>,<s>) : " << avgH[h] <<"; "<<
avgB[h] <<"; "<< avgS[h] << endl;
        }

```

```

Double_t ar[maxg+1];
for (int s = 0; s <= maxg; s++){
    ar[s]=s;
}
///  

TCanvas *C;
TGraph *g1, *g2, *g3, *gr1, *gr2, *gr3;

```

```

C = new TCanvas("vC1","square",200,10,600,1000);

```

```

g1 = new TGraph(maxg+1,ar,avgH);
g1->SetTitle("Correlation between <nh> <nb> <ns> from Ng");
g1->GetXaxis()->SetTitle("Ng (10B->2He)");
TF1 *fitFunc = new TF1("fitFunc", "[0]+[1]*x",0,10);
fitFunc->SetParameters(1.0,1.0);
fitFunc->SetLineStyle(1);
fitFunc->SetLineColor(kBlack);
g1->Fit(fitFunc);
g1->SetMarkerStyle(20);
g1->Draw("APL");
g1->GetXaxis()->SetRangeUser(0, 10);
g1->GetYaxis()->SetRangeUser(0, 20);

```

```

//C->cd(2);
g2 = new TGraph(maxg+1,ar,avgB);
TF1 *fitFunc2 = new TF1("fitFunc2", "[0]+[1]*x",0,10);
fitFunc2->SetParameters(1.0,1.0);
fitFunc2->SetLineStyle(9);
fitFunc2->SetLineColor(kBlack);
g2->Fit(fitFunc2);
g2->SetMarkerStyle(23);
g2->Draw("PL same");
//g2->GetXaxis()->SetRangeUser(0, 10);
//g2->GetYaxis()->SetRangeUser(0, 20);

```

```

//C->cd(3);
g3 = new TGraph(maxg+1,ar,avgS);
TF1 *fitFunc3 = new TF1("fitFunc3", "[0]+[1]*x",0,10);
fitFunc3->SetParameters(1.0,1.0);
fitFunc3->SetLineStyle(10);

```

```

fitFunc3->SetLineColor(kBlack);
g3->Fit(fitFunc3);
g3->SetMarkerStyle(25);
//g3->GetXaxis()->SetRangeUser(0, 10);
//g3->GetYaxis()->SetRangeUser(0, 20);
g3->Draw("PL same");

TLegend *legend = new TLegend(0.7, 0.7, 0.9, 0.9);
legend->AddEntry(g1, "- <nh> ", "p");
legend->AddEntry(g2, "- <nb> ", "p");
legend->AddEntry(g3, "- <ns> ", "p");
legend->Draw();
TLegend *legend2 = new TLegend(0.7, 0.7, 0.9, 0.9);
legend2->AddEntry(fitFunc, " - fit function for <nh> from Ng", "l");
legend2->AddEntry(fitFunc2, " - fit function for <nb> from Ng", "l");
legend2->AddEntry(fitFunc3, " - fit function for <ns> from Ng", "l");
legend2->Draw();

return 0;
}

```



## ПРИЛОЖЕНИЕ Л

### Программный код для построения корреляции множественности от ns для канала $^{10}\text{B} \rightarrow 2\text{He}$

```
#include <iostream>
#include <fstream>
#include <vector>
#include "Riostream.h"
#include "math.h"
#include "TImage.h"
#include "TCanvas.h"
#include "TArrayD.h"
#include "TROOT.h"
#include "TColor.h"
#include "TAttImage.h"
#include "TPluginManager.h"

using namespace std;

int fitNg() {
    ifstream file("B-01-18.dat");           // открываем файл
    для чтения
    vector<int> ns, nb, ng, nh;             // объявляем
    векторы для хранения значений
    int n2, n1;                             //
    переменные для считывания из файла
    int id, temp_nb, temp_ng, temp_ns, temp_n2, temp_n1, temp_n3, temp_n4; //
    временные переменные для считывания из файла

    if (file) {                             // если файл
    успешно открыт
        while (file >> id >> temp_nb >> temp_ng >> temp_ns >> temp_n4 >> temp_n3
        >> temp_n2 >> temp_n1)
        {
            // считываем данные из файла
            if (temp_n2 == 2 && temp_n1 == 0) { // если условие выполняется
                nb.push_back(temp_nb);         // добавляем значения в
                векторы
                ng.push_back(temp_ng);
                ns.push_back(temp_ns);
                nh.push_back(temp_nb + temp_ng);
            }
        }
        file.close();                       //
    закрываем файл
    }

    //////////////////////////////////////
    //вычисление среднего значения через вероятность
    int maxb = *max_element(nb.begin(), nb.end()); // находим максимальные
    значения
    int maxg = *max_element(ng.begin(), ng.end());
```

```

    int maxs = *max_element(ns.begin(), ns.end());
    int maxh = *max_element(nh.begin(), nh.end());

    double probS[maxg+1][maxs+1]; // объявление массива
    double probH[maxg+1][maxh+1];
    double probB[maxg+1][maxb+1];

    for (int h = 0; h <= maxg; h++)
    {
        cout<<"////////// " <<endl;
        cout << "ng=" << h << ":" << endl;
        cout<<" " <<endl;

        for (int s = 0; s <= maxs; s++) { // проходим по значениям ns
            probS[h][s] = 0;
            float probability = 0; //
переменная для подсчета вероятности
            float p=0;
            int total_count = 0;
            int count = 0; //
счетчик для подсчета количества событий
            for (int i = 0; i < ns.size(); i++) {

                if (ng[i] == h) { // если nh равно
текущему значению h
                    total_count++;
                // увеличиваем счетчик событий
                if (ns[i] == s) probability++; // если ns
равно s=0, увеличиваем счетчик вероятности
            }
        }
        if(total_count==0){
            p=0;
        }else
        p=probability / total_count;
        probS[h][s] = p;
        //cout << prob[h][s] << " ";
        cout << "ns=" << s << " for probability p=" << p << ";" << endl; // выводим
результаты
    }
    cout<<" " <<endl;

    for (int g = 0; g <= maxh; g++) { // проходим по значениям
ng
        probH[h][g] = 0;
        float probability = 0; //
переменная для подсчета вероятности
        float p=0; // переменная для
подсчета вероятности

```

```

        int total_count = 0; //
счетчик для подсчета количества событий
        for (int i = 0; i < nh.size(); i++) {

            if (ng[i] == h) { // если nh равно
текущему значению h

                total_count++;
                // увеличиваем счетчик событий
                if (nh[i] == g) probability++; // если ns
равно s, увеличиваем счетчик вероятности
            }
        }

        if(total_count==0){
            p=0;
        }else
            p=probability / total_count;
        probH[h][g] = p;
        cout << "nh=" << g << " for probability p=" << p << ";" << endl; // выводим
результаты
    }
    cout<<" "<<endl;

    for (int b = 0; b <= maxb; b++) { // проходим по значениям
nb

        probB[h][b] = 0;
        float probability = 0; //
переменная для подсчета вероятности
        float p=0; // переменная для
подсчета вероятности
        int total_count = 0; // счетчик
для подсчета количества событий
        for (int i = 0; i < nb.size(); i++) {

            if (ng[i] == h) { // если nh равно
текущему значению h

                total_count++;
                // увеличиваем счетчик событий
                if (nb[i] == b) probability++; // если ns
равно s=0, увеличиваем счетчик вероятности
            }
        }

        if(total_count==0){
            p=0;
        }else
            p=probability / total_count;
        probB[h][b] = p;
        cout << "nb=" << b << " for probability p=" << p << "; Ns = " << probability
<< "; all = " << total_count << endl; // выводим результаты
    }
    cout<<" "<<endl;

```

```

    }
    cout<<" "<<endl;

    double avgH[maxg+1];
    for (int h = 0; h <= maxg; h++) {
        double avg = 0;
        for (int g = 0; g <= maxh; g++){
            avg += g * probH[h][g];
        }
        avgH[h] = avg;
    }
    double avgS[maxg+1];
    for (int h = 0; h <= maxg; h++) {
        double avg = 0;
        for (int s = 0; s <= maxs; s++){
            avg += s * probS[h][s];
        }
        avgS[h] = avg;
    }
    double avgB[maxg+1];
    for (int h = 0; h <= maxg; h++) {
        double avg = 0;
        for (int s = 0; s <= maxb; s++){
            avg += s * probB[h][s];
        }
        avgB[h] = avg;
    }

    for (int h = 0; h <= maxg; h++) {
        cout << "g= " << h << " (<h>,<b>,<s>) : " << avgH[h] <<"; "<< avgB[h] <<";
"<< avgS[h] << endl;
    }

    Double_t ar[maxg+1];
    for (int s = 0; s <= maxg; s++){
        ar[s]=s;
    }
    ///*
    TCanvas *C;
    TGraph *g1, *g2, *g3, *gr1, *gr2, *gr3;

    C = new TCanvas("vC1","square",200,10,600,1000);

    g1 = new TGraph(maxg+1,ar,avgH);
    g1->SetTitle("Correlation between <nh> <nb> <ns> from Ng");
    g1->GetXaxis()->SetTitle("Ng (10B->2He)");
    TF1 *fitFunc = new TF1("fitFunc", "[0]+[1]*x",0,10);
    fitFunc->SetParameters(1.0,1.0);
    fitFunc->SetLineStyle(1);
    fitFunc->SetLineColor(kBlack);

```

```

g1->Fit(fitFunc);
g1->SetMarkerStyle(20);
g1->Draw("APL");
g1->GetXaxis()->SetRangeUser(0, 10);
g1->GetYaxis()->SetRangeUser(0, 20);

//C->cd(2);
g2 = new TGraph(maxg+1,ar,avgB);
TF1 *fitFunc2 = new TF1("fitFunc2", "[0]+[1]*x",0,10);
fitFunc2->SetParameters(1.0,1.0);
fitFunc2->SetLineStyle(9);
fitFunc2->SetLineColor(kBlack);
g2->Fit(fitFunc2);
g2->SetMarkerStyle(23);
g2->Draw("PL same");
//g2->GetXaxis()->SetRangeUser(0, 10);
//g2->GetYaxis()->SetRangeUser(0, 20);

//C->cd(3);
g3 = new TGraph(maxg+1,ar,avgS);
TF1 *fitFunc3 = new TF1("fitFunc3", "[0]+[1]*x",0,10);
fitFunc3->SetParameters(1.0,1.0);
fitFunc3->SetLineStyle(10);
fitFunc3->SetLineColor(kBlack);
g3->Fit(fitFunc3);
g3->SetMarkerStyle(25);
//g3->GetXaxis()->SetRangeUser(0, 10);
//g3->GetYaxis()->SetRangeUser(0, 20);
g3->Draw("PL same");

TLegend *legend = new TLegend(0.7, 0.7, 0.9, 0.9);
legend->AddEntry(g1, "- <nh> ", "p");
legend->AddEntry(g2, "- <nb> ", "p");
legend->AddEntry(g3, "- <ns> ", "p");
legend->Draw();
TLegend *legend2 = new TLegend(0.7, 0.7, 0.9, 0.9);
legend2->AddEntry(fitFunc, " - fit function for <nh> from Ng", "l");
legend2->AddEntry(fitFunc2, " - fit function for <nb> from Ng", "l");
legend2->AddEntry(fitFunc3, " - fit function for <ns> from Ng", "l");
legend2->Draw();

return 0;
}

```

## ПРИЛОЖЕНИЕ М

### Заключение о результатах проверки магистерской диссертации на отсутствие плагиата

**РУКОНТЕКСТ**  
text.rucont.ru

#### Справка о проверке на наличие заимствований

Имя файла: Анализ особенностей периферических взаимодействий легких релятивистских ядер в ядерной эмульсии.docx

Автор: Кашанская Ольга Николаевна

Заглавие: Анализ особенностей периферических взаимодействий легких релятивистских ядер в ядерной эмульсии

Год публикации: 2023

Комментарий: Не указан

Коллекции: Интернет 2.0, Научные статьи 2.0, Русскоязычная Википедия, Англоязычная Википедия, Коллекция Энциклопедий, Библиотека Либрусек, Коллекция КФУ, Коллекция ПГУТИ, Репозиторий открытого доступа СПб гос. ун-та, Научная электронная библиотека "КиберЛенинка", Авторефераты ВАК, Диссертации ВАК, Готовые рефераты, ФИПС. Изобретения, ФИПС. Полезные модели, ФИПС. Промышленные образцы, Коллекция Руконт, Библиотека им. Ушинского, Готовые рефераты (часть 2), Открытые научные источники, Правовые документы I, Правовые документы II, Правовые документы III, Собрание законодательства Российской Федерации



#### Результат проверки

Оценка оригинальности документа: **90%**

Оригинальные фрагменты: 90,03%

Обнаруженные заимствования: 7,73%

Цитирование: 2,24%

