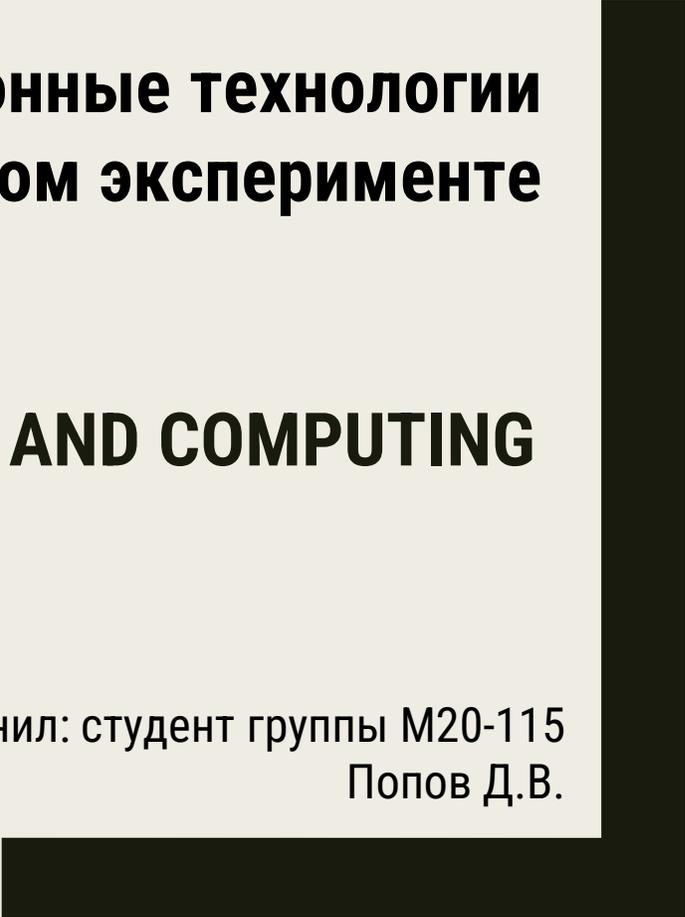




**Информационные технологии
в физическом эксперименте**

NEUTRINO EXPERIMENTS AND COMPUTING

Выполнил: студент группы М20-115
Попов Д.В.



Москва 2021

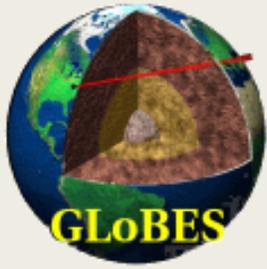
Введение

Таблица 1. Характерные значения длины базы L и энергии нейтрино E_ν для различных источников нейтрино и примеры соответствующих экспериментов

	$L, \text{ м}$	$E_\nu, \text{ МэВ}$	Примеры
Солнечные	10^{10}	1 – 10	SAGE, GALLEX, GNO, SNO, KamLAND
Атмосферные	$10^4 - 10^7$	$10^2 - 10^5$	Super-Kamiokande, ANTARES, IceCube
Ускорительные	SBL: $10 - 10^2$ LBL: $10^5 - 10^6$	$10^3 - 10^4$	LSND, MiniBooNE, T2K, NOvA, DUNE, P20
Реакторные	SBL: $10^2 - 10^3$ LBL: $10^4 - 10^5$	1 – 10	RENO, Daya Bay, KamLAND, JUNO

Открытые вопросы:

- 1) Иерархия масс нейтрино – прямая или обратная?
- 2) Природа массы нейтрино – Дираковская или Майорановская?
- 3) CP-нарушение в лептонном секторе Стандартной модели – $\delta_{CP} = 0$?
- 4) Существуют ли стерильные состояния нейтрино? Если да, какую роль они сыграли в формировании Вселенной?
- 5) Неунитарность матрицы PMNS, нестандартные взаимодействия, Лоренц-нарушение и другие проявления новой физики.



GLOBES

программный пакет для моделирования осцилляционных экспериментов с длинной базой

- **AEDL (Abstract Experiment Definition Language)** – простой способ задания экспериментальной установки;
- Монте-Карло генерация, статистический анализ данных (алгоритмы оптимизации χ^2 , функции правдоподобия и т.п.);
- Инструментарий для моделирования нестандартной физики – стерильные состояния нейтрино, неунитарность матрицы PMNS и т.п.
- Готовая библиотека нейтринных экспериментов: T2K, NO ν A, Double Chooz, RENO и т.д.

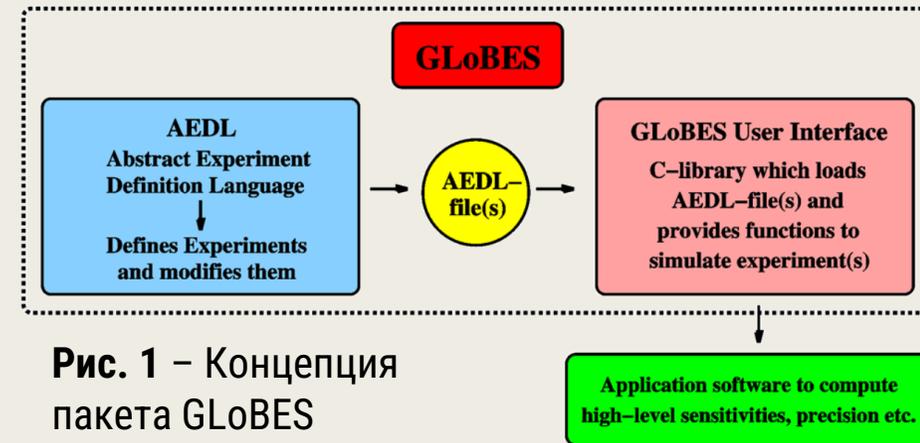


Рис. 1 – Концепция пакета GLOBES

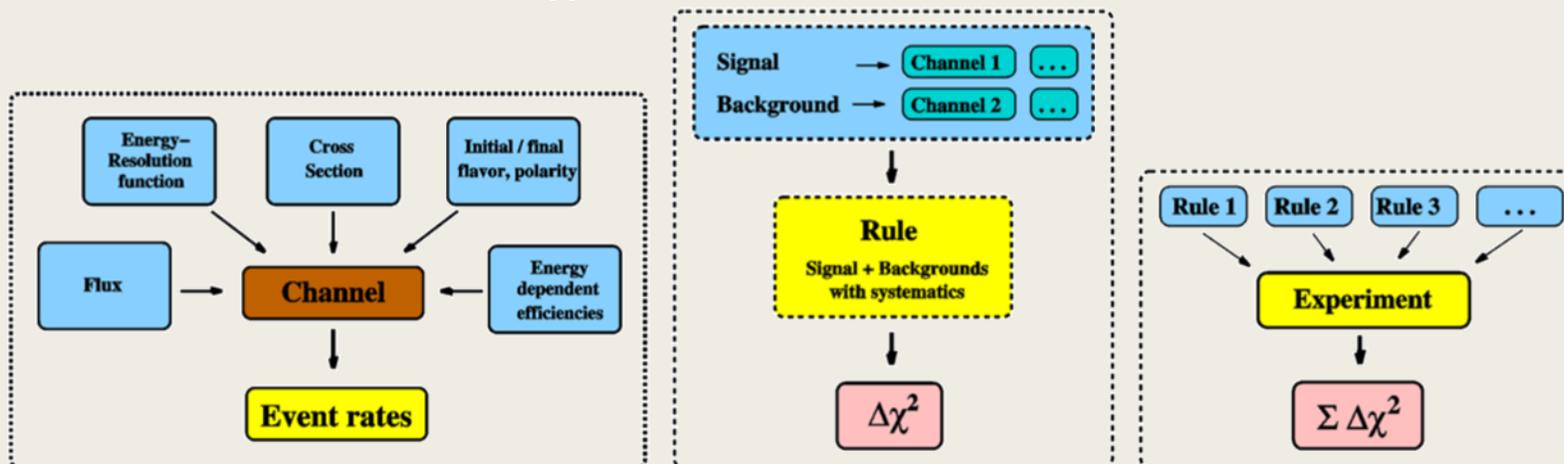


Рис. 2 – Наиболее важные компоненты AEDL: *channels, rules, experiments*

Реакторные эксперименты: моделирование потока антинейтрино

1. **Метод прямого суммирования (*ab initio*)** – использование ядерных баз данных (ENDF, GEFY, JEFF, JENDL) для построения спектра антинейтрино; необходимо организовать запрос в ЯБД для получения информации о всех возможных каналах распада всех возможных осколков деления основных изотопов: ^{235}U , ^{238}U , ^{239}Pu , ^{241}Pu (порядка 10^4 ветвей);
2. **Метод конверсии** – расчет спектров антинейтрино по имеющимся кумулятивным бета-спектрам (группа ILL, 80-ые годы) основных изотопов;

Для расчета потоков антинейтрино методом конверсии на данный момент существует 3 основные модели и соответствующих этим моделям ПО: **Huber-Mueller** (2011), **HKNS** (2019) и **НИЦ КИ** (2021).

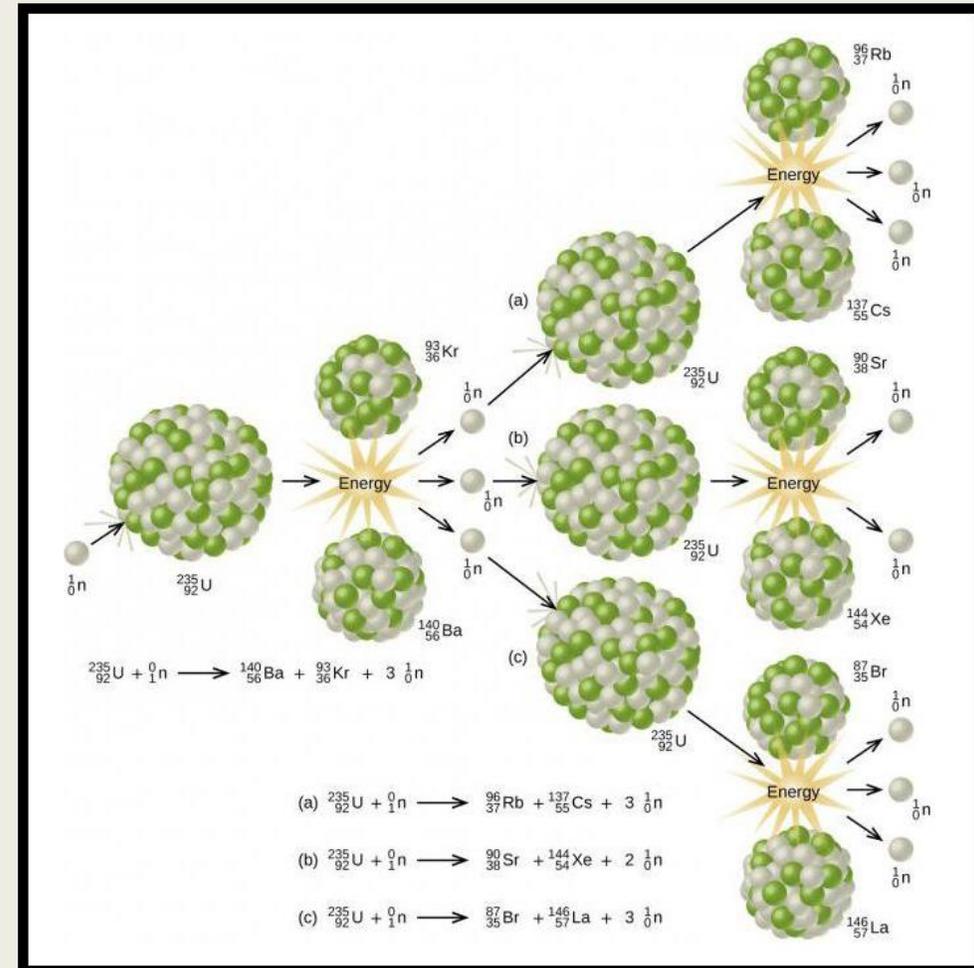


Рис. 3 – Возможная схема деления ^{235}U и его осколков

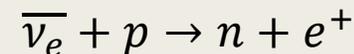
Daya Bay

- осцилляционный реакторный эксперимент (Китай) с базой $L \approx 1$ км



Рис. 4 – Карта эксперимента Daya Bay

- Источник антинейтрино – 6 атомных реакторов (тепловая мощность каждого порядка 3 ГВт);
- 8 жидкосцинтилляционных детекторов, расположенных в 3 экспериментальных залах;
- Регистрация антинейтрино по реакции обратного бета-распада



2 сигнала: от аннигиляции позитрона с электроном и от захвата нейтрона протоном среды

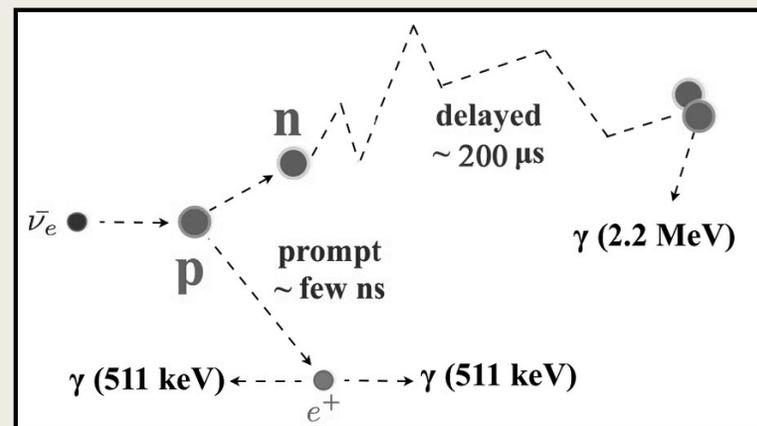


Рис. 5 – Обратный бета-распад

Daya Bay: вычислительная среда

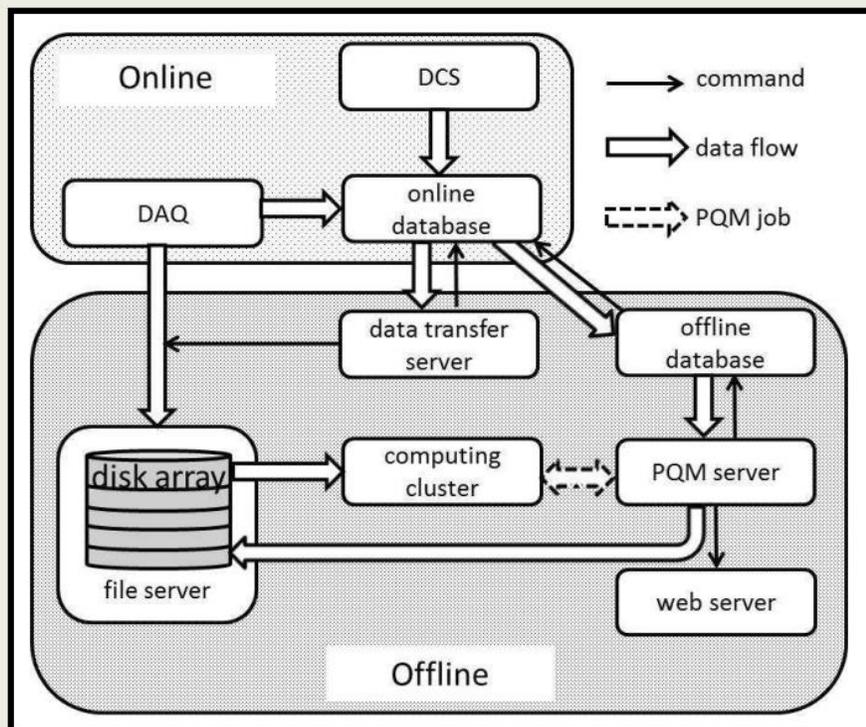


Рис. 6 - Принципиальная схема автономной вычислительной среды в Daya Bay;
DCS – система управления детектором
DAQ – система сбора данных
PQM – система обработки данных
PBS – Portable Batch System
PQM – Performance Quality Monitoring system

11 серверов:

Файловый сервер, сервер передачи данных, сервер автономной базы данных, веб-сервер, 5 ферм пользователей;
 Общий объем - 25 ТБ, из них: 17 ТБ для хранения необработанных данных, 5 ТБ для пользователей, 2 ТБ для автономного ПО, 1 ТБ для ROOT-файлов;

Во время сбора физических данных события из разных экспериментальных залов отдельно собираются в формате необработанных данных и сохраняются в файлах на сетевом дисковом массиве системой сбора данных (*DAQ*). Общее количество файлов необработанных данных за каждый день составляет около 320, примерно 1 ГБ объема на файл.

DAQ создает файл необработанных данных, информация, включая метки начала и окончания файла, номер запуска, номер файла и т.п. сбрасывается в **online database** для создания новой записи; записываются значения напряжений ФЭУ, температура детектора, уровни воды и т.д. (эта информация передается из *DCS*). Необработанные данные копируются на **offline database** после того, как сервер передачи данных обнаружит новую запись в онлайн базе-данных (копии хранятся 1 месяц). Помимо этого, в **offline database** хранится информация о калибровке детектора.

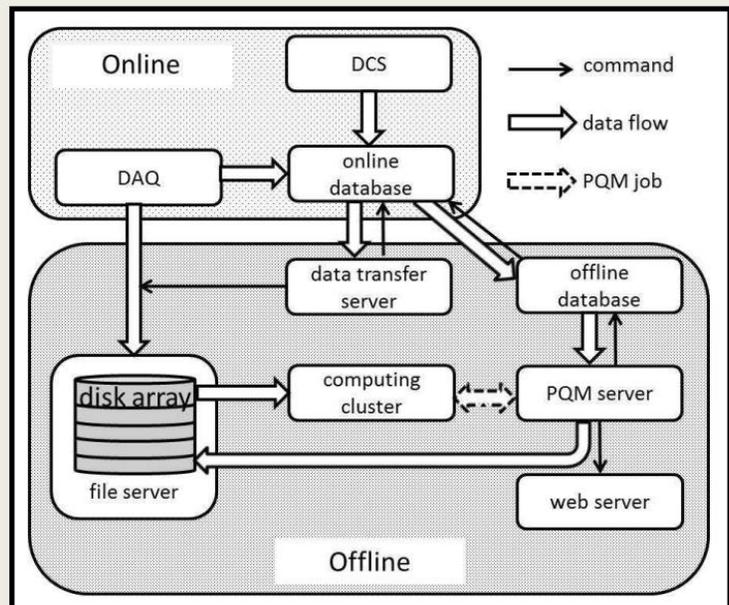


Рис. 6 - Принципиальная схема автономной вычислительной среды в Daya Bay;
DCS – система управления детектором
DAQ – система сбора данных
PQM – система обработки данных
PBS – Portable Batch System
PQM – Performance Quality Monitoring system

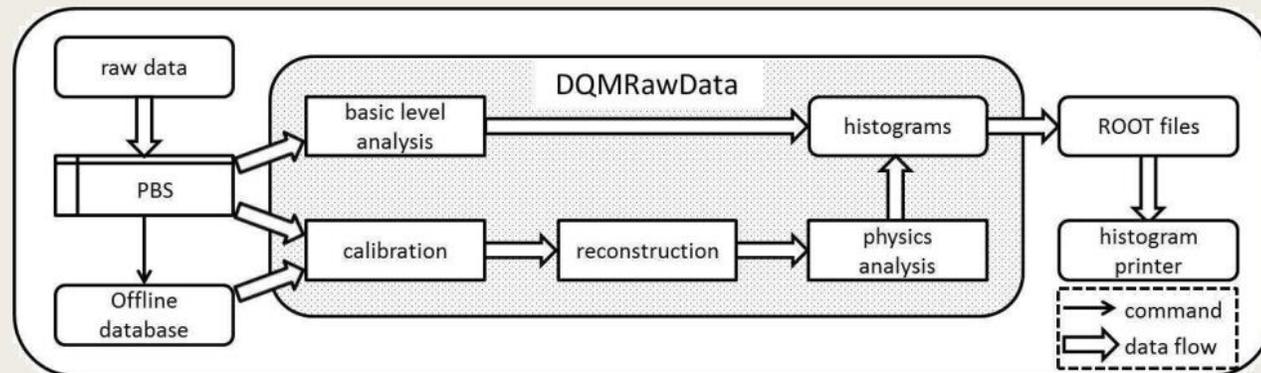


Рис. 7 - Поток данных PQM

Для управления потоком данных **PQM** был разработан сценарий управления на языке Python, который выполняется в фоновом режиме на **сервере PQM**. Когда **DAQ** закрывает файл необработанных данных, **offline database** будет обновлена новой записью, скопированной из **online database**. Копия файла будет передана на файловый сервер менее чем за 5 минут, при условии, что размер ≈ 1 Гб. Когда передача данных завершена, статус этого файла устанавливается как TRANSFERRED как в **online database**, так и в **offline database**.

Управляющий скрипт делает запросы в **offline database** с фиксированным интервалом времени (10 секунд). Если найдена новая запись со статусом TRANSFERRED в обеих базах, задание на обработку нового файла отправляется в **PBS**. **PQM** использует данные о калибровке из **offline database**, запускает алгоритмы анализа и реконструкции, генерирует гистограммы. Все гистограммы сбрасываются в ROOT-файл на этапе завершения алгоритмов, который затем объединяется с накопленным ROOT-файлом для одного и того же запуска. После этого выполняется принтер гистограмм на языке C++, принимающий объединенный ROOT-файл в качестве входных данных для
 Управляющий сценарий также проверяет, завершен ли алгоритм, проверяя создание текстового файла. Для завершенных алгоритмов сценарий управления передает информацию на диск PQM для веб-отображения. Когда вся процедура завершается и все соответствующие файлы необработанных данных обрабатываются, сценарий управления сохраняет накопленный ROOT-файл на диске PQM для постоянного хранения.



NOvA

ускорительный эксперимент с пучком нейтрино из Fermilab со средней энергией $\langle E_\nu \rangle \approx 2$ ГэВ

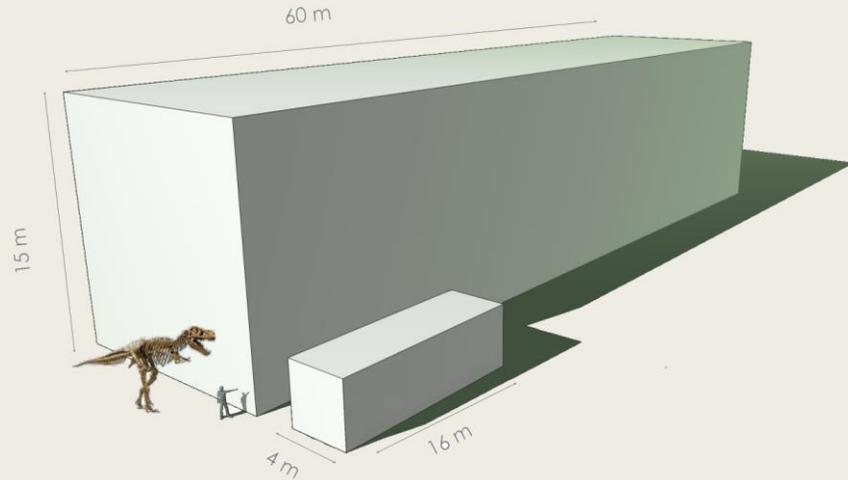
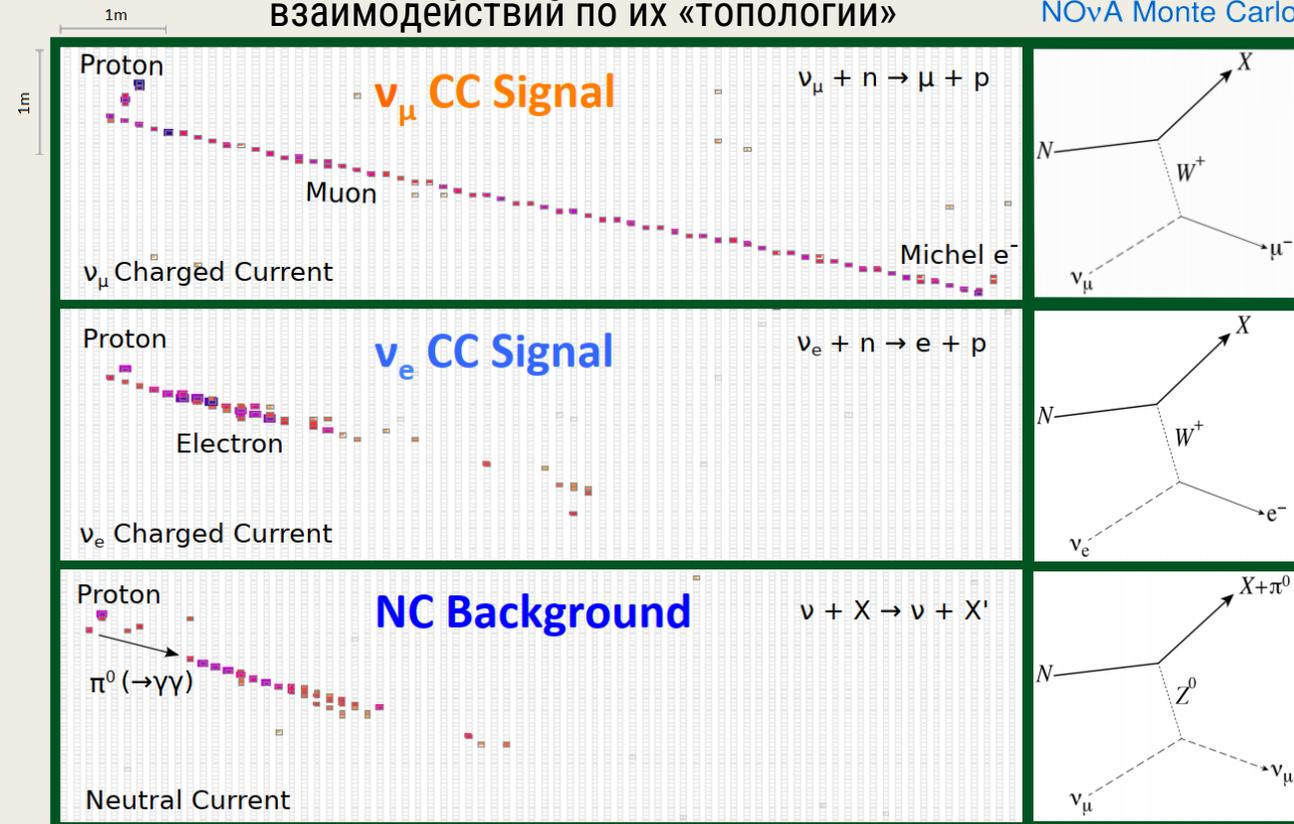


Рис. 8 – масштаб детекторов NOvA;
 ближний детектор: 330 т, 1 км от источника
 дальний детектор: 14 кт, 810 км от источника

Устройство обоих детекторов одинаково – они состоят из поливинилхлоридных ячеек, заполненных жидким сцинтиллятором, а световые импульсы от них собираются специальным оптоволоком. Активность в ячейках регистрируется в виде сверху и сбоку. Ближний детектор находится под землей, дальний – на поверхности.

Рис. 9 – идентификация событий и взаимодействий по их «топологии»

NOvA Monte Carlo





NOνA

Способы восстановления энергии нейтрино:

1) *Калориметрический*: просуммировать энергии ячеек по ячейкам и умножить на масштабный коэффициент, учитывающий потерянную энергию и мертвый слой детектора (масштабный коэффициент вычисляется из МК-генерации):

$$E_\nu = k \sum_j E_j$$

2) *Кинематический*: энергия нейтрино связана с адронной и электромагнитными энергиями кинематической квадратичной зависимостью:

$$E_\nu = f(E_{had}, E_{EM}), \quad E_{had} = \sum_{j,had} E_j, \quad E_{EM} = \sum_{j,EM} E_j$$

3) С использованием сверточной нейронной сети (CNN)

Энергия нейтрино зависит от «топологии» (сигнатуры) события

Дополнительно: Convolutional Neural Network (CNN)

Задача классификации изображений – это приём начального изображения и вывод его класса (кошка, собака и т.д.) или группы вероятных классов, которая лучше всего характеризует изображение.

Рис. 10 – Свертка матриц

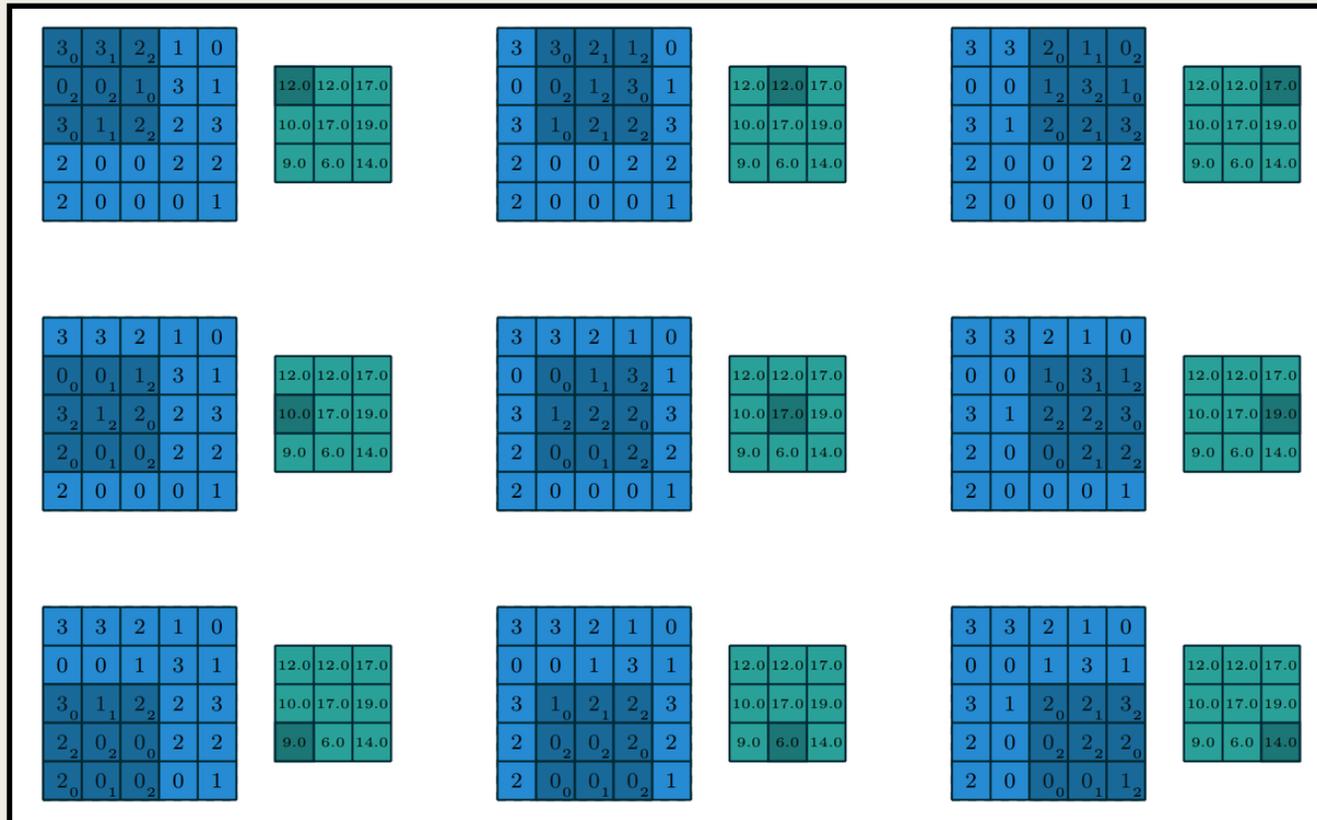
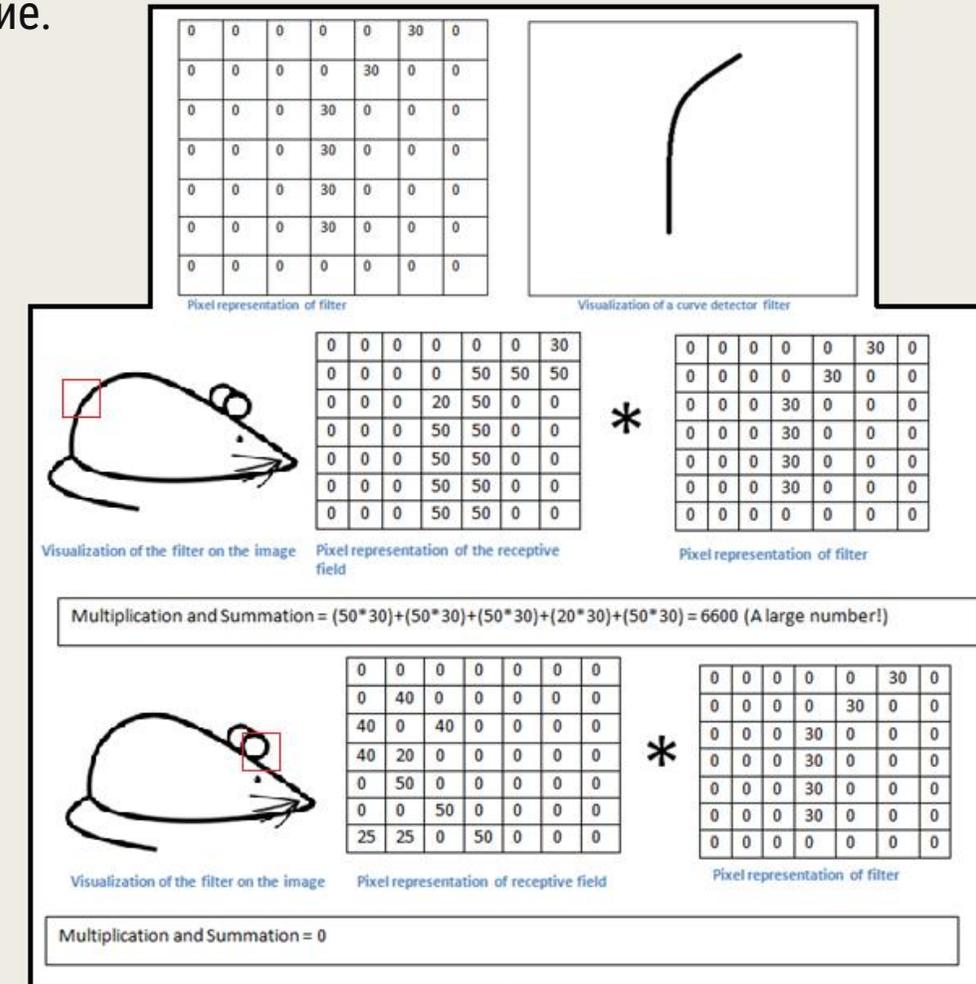


Рис. 11 – Иллюстрация работы CNN





NOvA

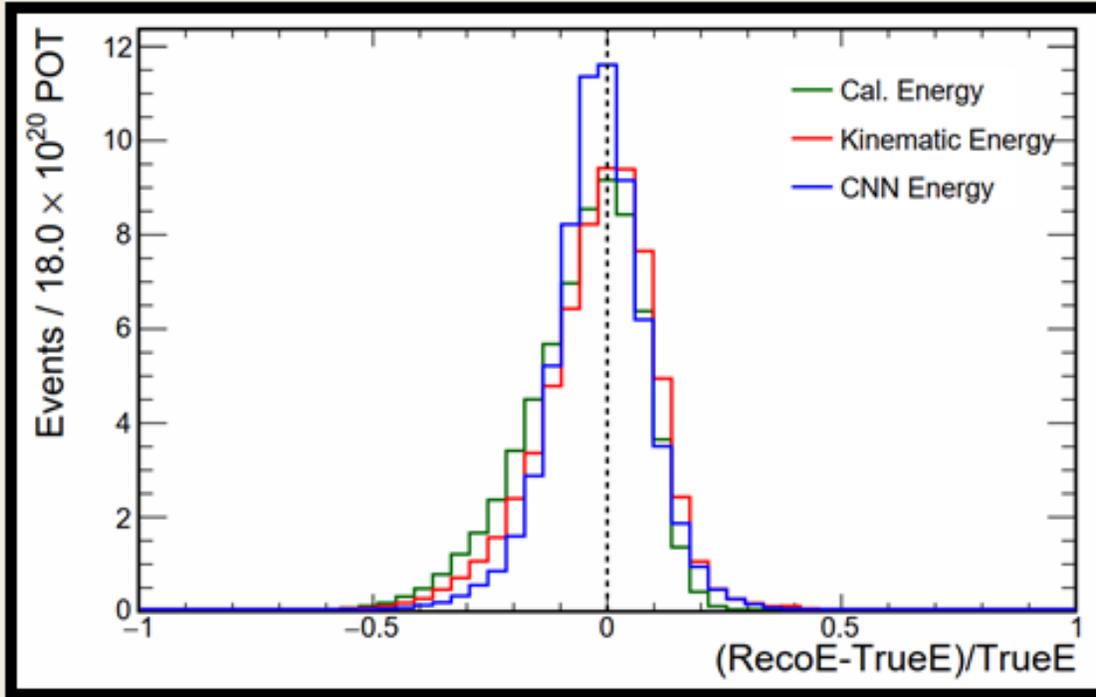


Рис. 12 – Распределение МК-событий по разности между восстановленной и истинной энергиями, нормированной на истинную энергию

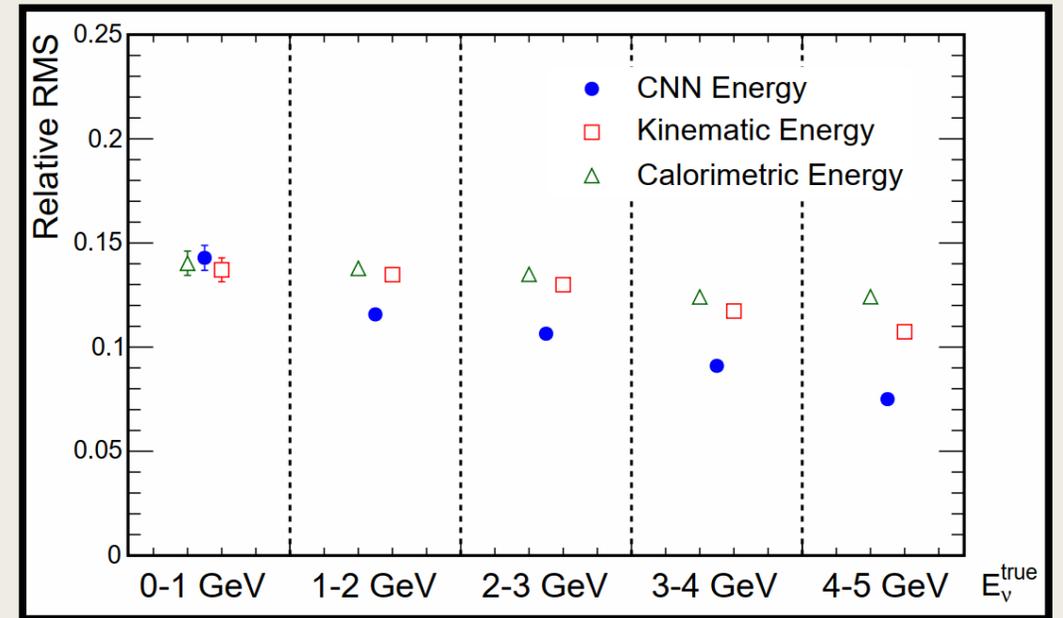
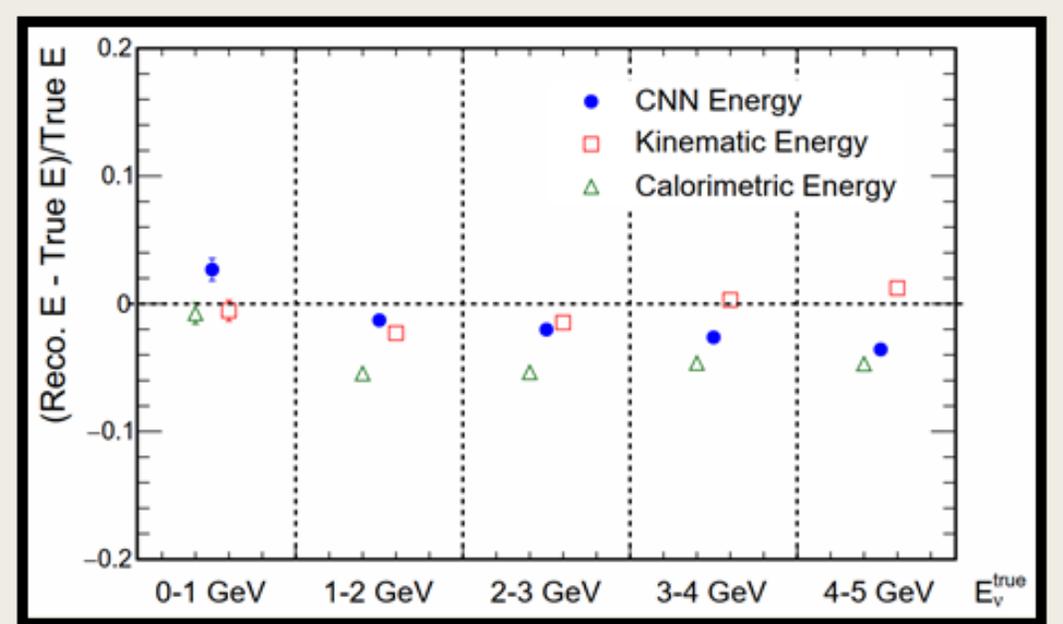


Рис. 13 – Среднее (сверху) и относительное среднеквадратичное (снизу) распределения соотношений разностей между восстановленной и истинной энергиями (Монте-Карло)

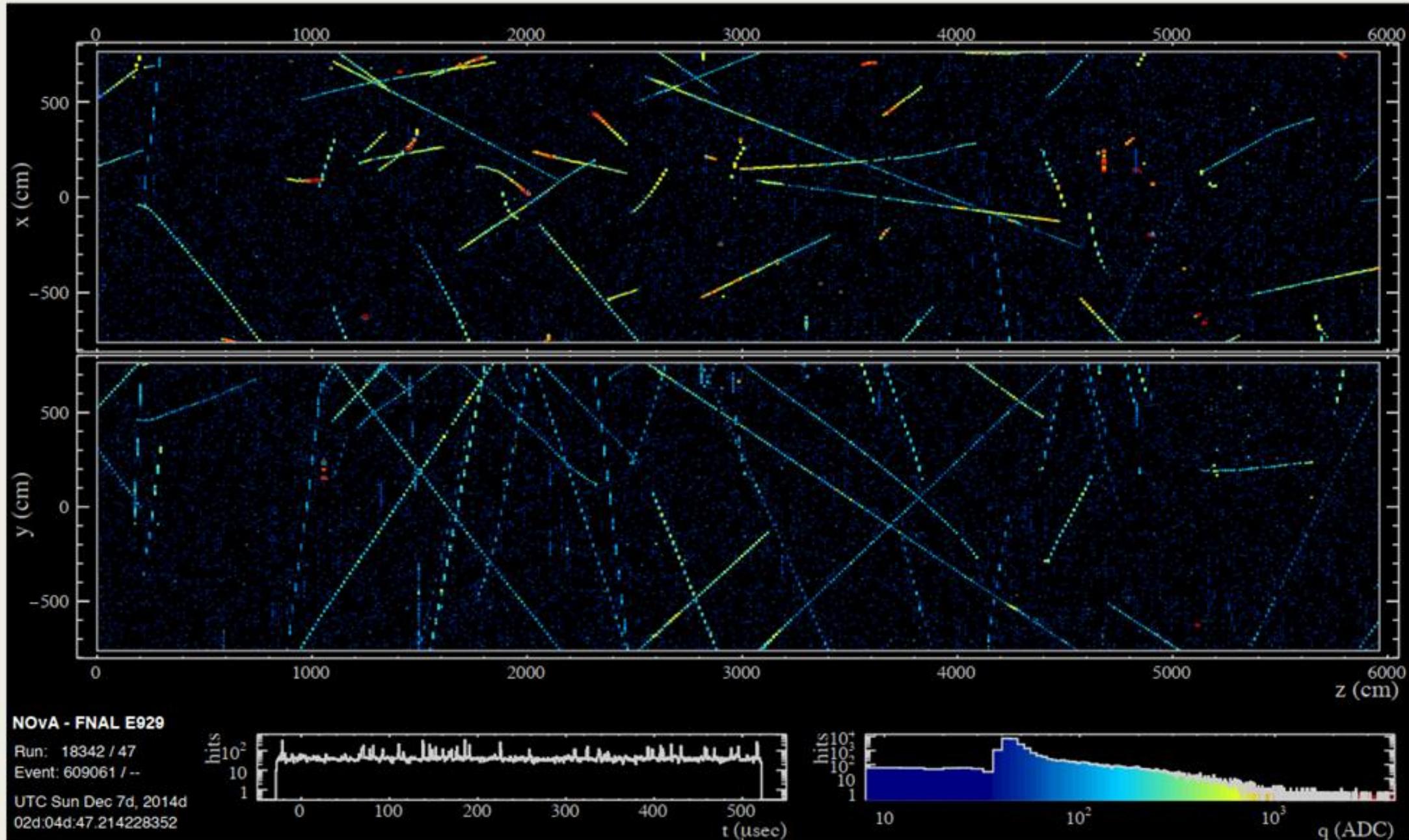


Рис. 14 – Дисплей установки NO ν A

DUNE



эксперимент нового поколения с пучком нейтрино из Fermilab (+атмосферные, солнечные, астрофизические)

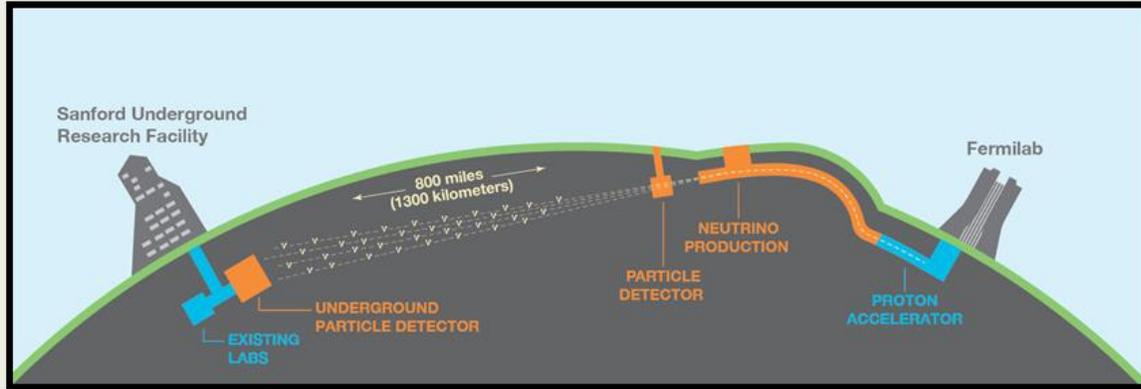


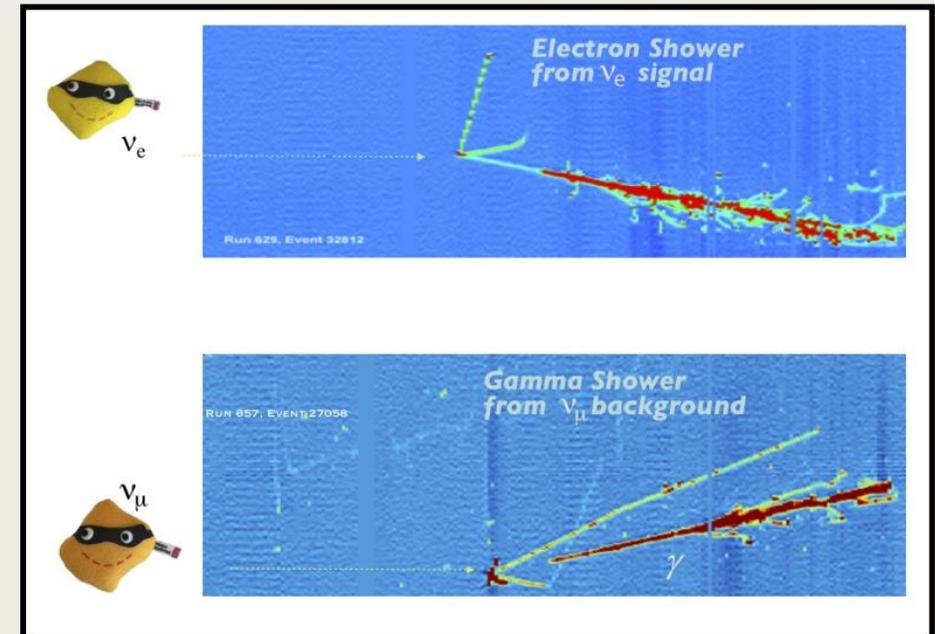
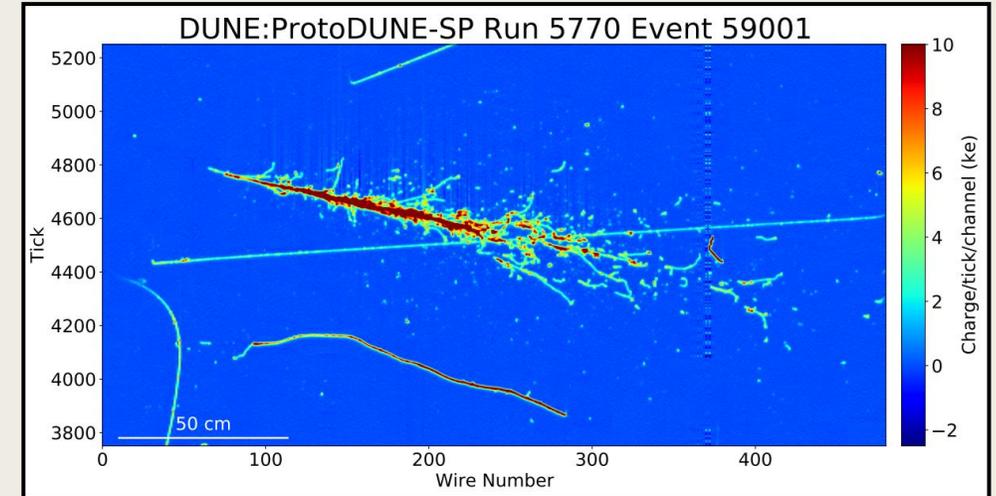
Рис. 15 – Схема эксперимента DUNE

Детектор DUNE: TPC (жидкий аргон) – 4 модуля по 10 кт (крупнейший на данный момент)

Объем данных – непрерывное считывание без сжатия – 1.5 ТБ/сек → схема отбора данных в реальном времени.

Алгоритмы машинного обучения + объем данных – вычислительная проблема. Возможное решение – **GPU** (как сопроцессоры).
Выполнение самой трудоемкой задачи – отслеживание и идентификация взаимодействия – ускоряется **в 17 раз**.

Оптимальная интеграция графических процессоров в рабочий процесс – **Services for Optimized Network Inference on Coprocessors (SONIC)**



Реконструкция начинается с применения процедуры **деконволюции** для восстановления исходных сигналов (отбрасывание шума и т.п.).

Деконволютированные формы сигналов используются для вычисления времени события и собранного заряда (энергии). После этого 2D-информация (X-Y), представленная совпадениями в каждой плоскости, объединяется с информацией из других плоскостей для восстановления трехмерных пространственных точек (**3D-реконструкция**).

Полученные данные передаются на следующий этап, в процессе которого выявляются ливни, вершины, треки. Они объединяются в иерархию родительско-дочерних узлов, которые в конечном итоге указывают на потенциальное взаимодействие нейтрино.

Последний модуль **EmTrackMichellD** представляет собой **алгоритм ML**, который классифицирует восстановленные сигналы по типу track-like, shower-like, или Michel electron-like.

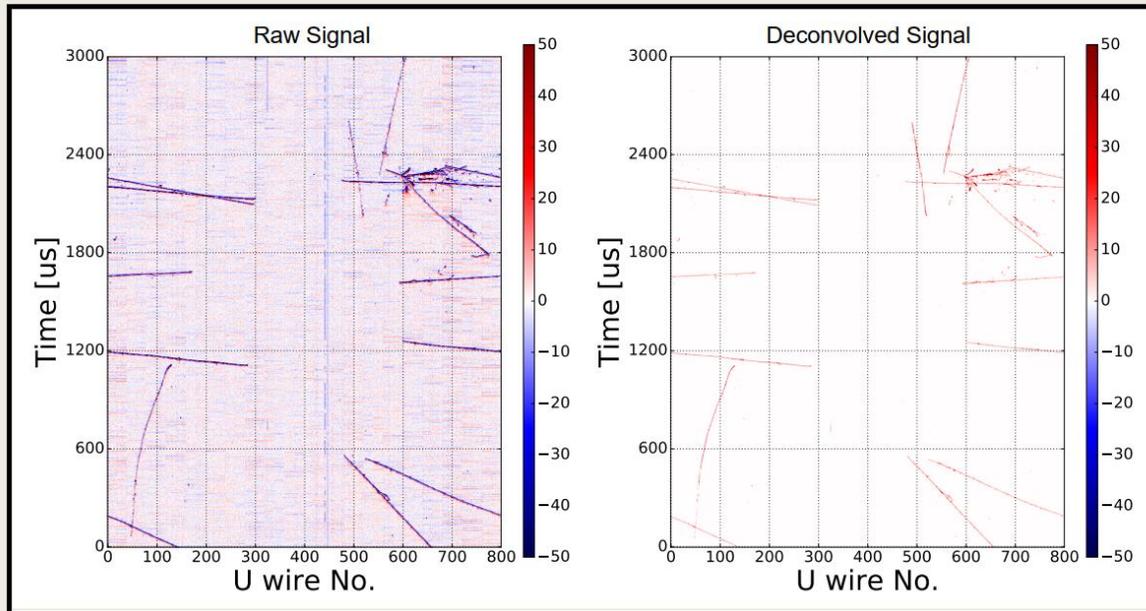


Рис. 17 – Сравнение сигналов до деконволюции и после

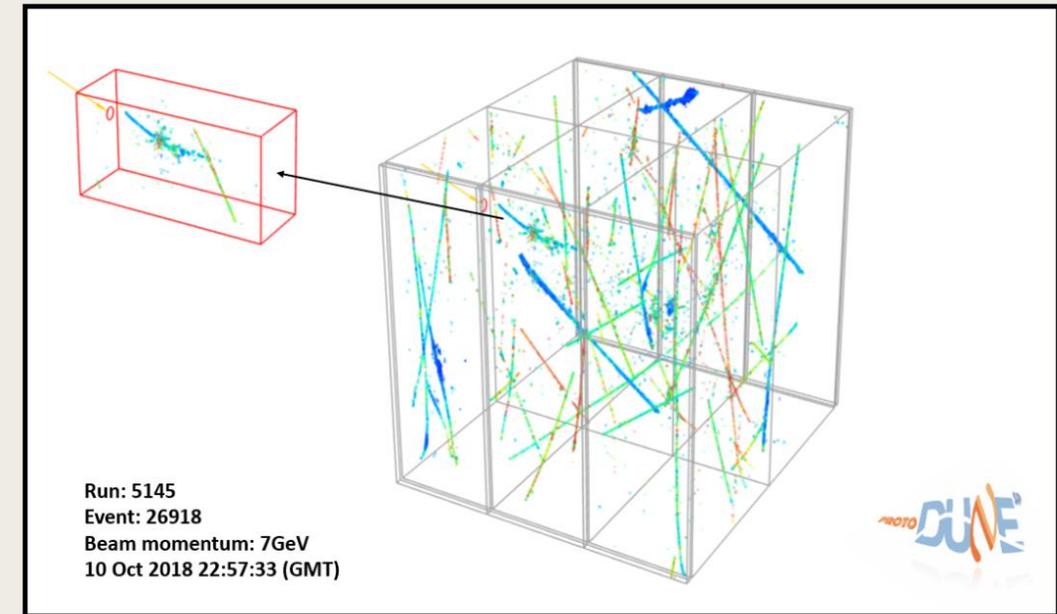
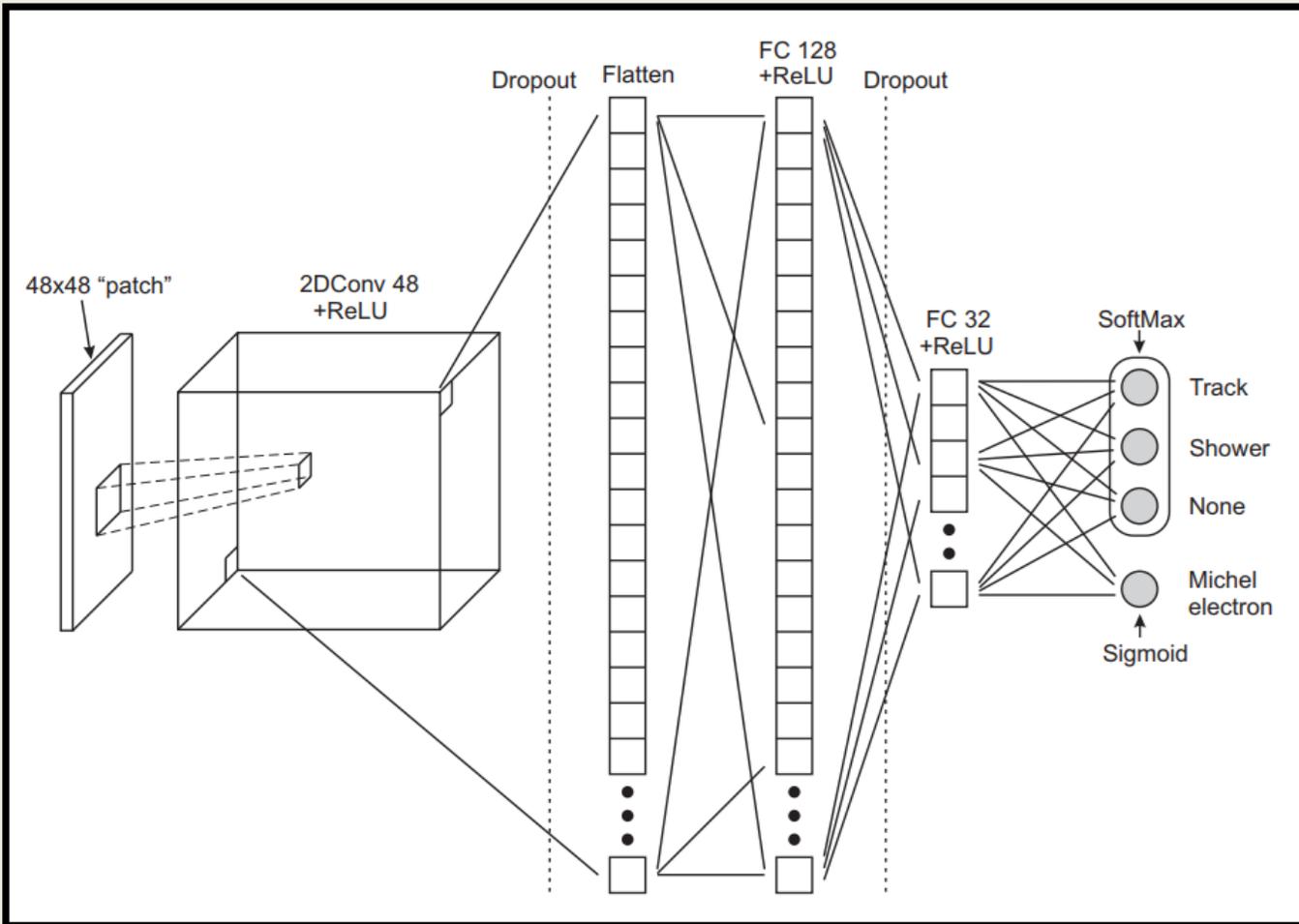


Рис. 18 – 3D-реконструкция

Архитектура CNN ProtoDUNE



Нейронная сеть, используемая модулем *EmTrackMichellD* цепи предварительной реконструкции ProtoDUNE, состоит из 2D-сверточного слоя, за которым следуют два полностью соединенных слоя (FC). Сверточный слой берет каждый из участков размером 48×48 пикселей (такое изображение в дальнейшем будем называть *patch*) и применяет к нему 48 отдельных сверток размером 5×5 для создания 48 соответствующих карт объектов размером 44×44 пикселя. Эти карты объектов затем подаются в первый слой, состоящий из 128 нейронов, соединенный со вторым слоем из 32 нейронов.

Функции активации применяются после сверточного слоя и каждого из двух слоев FC.

Рис. 15
Архитектура CNN, используемой в модуле *EmTrackMichellD* цепи реконструкции ProtoDUNE

Код реконструкции ProtoDUNE-SP основан на программной платформе **Larsoft C++**, которая предоставляет общий набор инструментов, используемых во многих экспериментах с нейтрино на основе LArTPC.

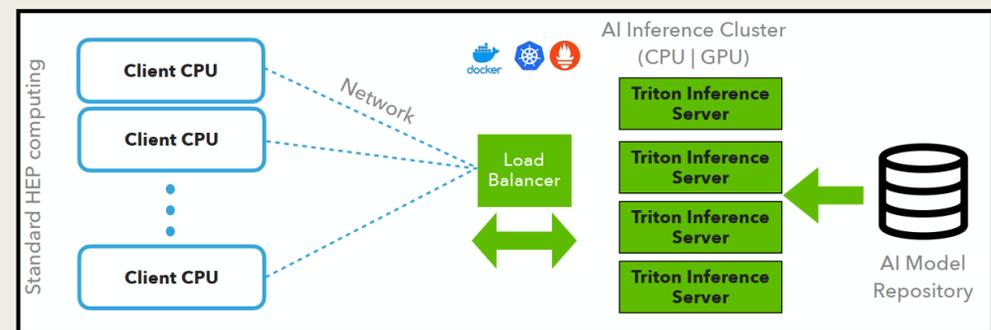
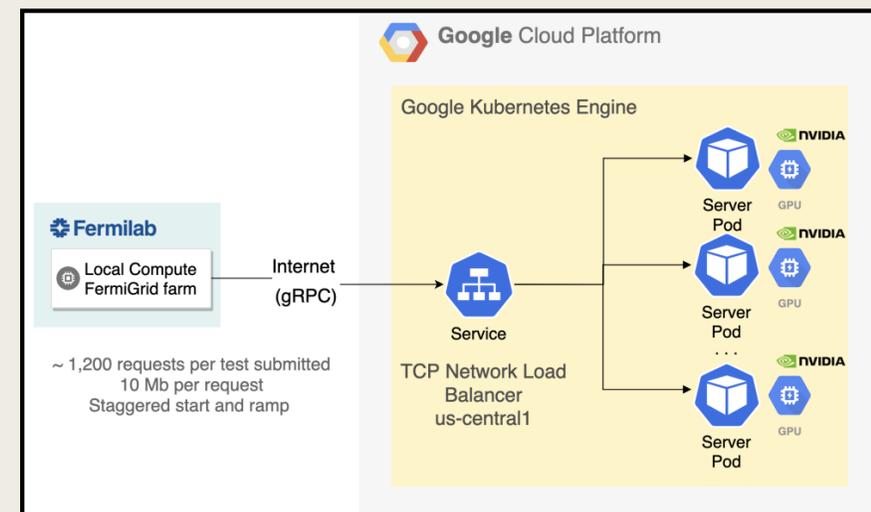
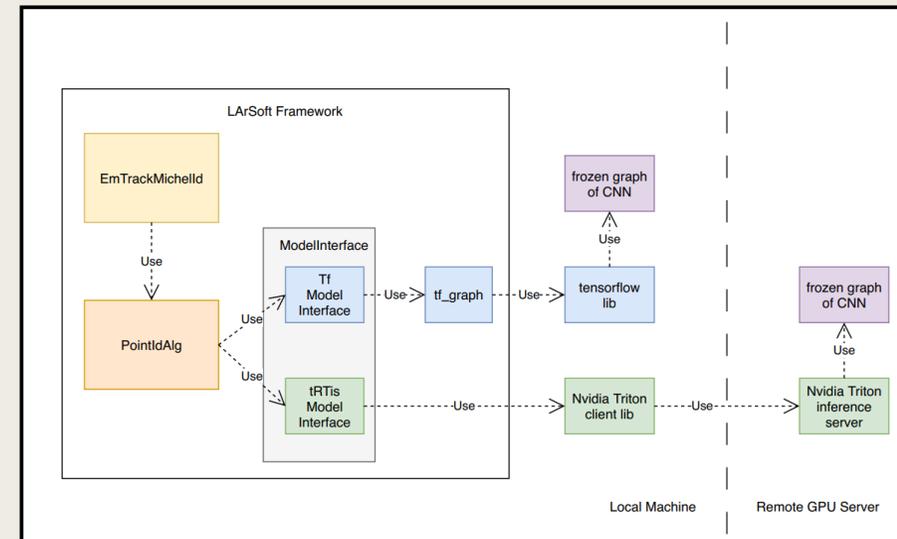
EmTrackMichellId передает обработанную информацию в **PointIdAlg**, который создает **patch** и выполняет задачу вывода его классификации.

SONIC - модель клиент-сервер, в которой аппаратное обеспечение сопроцессора, используемое для ускорения вывода CNN, отделено от пользователя ЦП и доступно как веб-сервис. Входные данные нейронной сети передаются по сети TCP/IP на графический процессор.

Класс **ModelInterface** в LARSOFT используется **PointIdAlg** для доступа к модели ML. Ранее этот вывод выполнялся локально на процессоре с использованием подкласса **TfModelInterface** интерфейса модели. Доступ к графическому процессору был реализован путем создания клиентского интерфейса C++, предоставляемого **сервером вывода Nvidia Triton**, в новом подклассе **ModelInterface**, называемом **tRTisModelInterface**. В этом новом подклассе исправления построенные с помощью **PointIdAlg**, помещаются в надлежащий формат и передаются на сервер GPU для обработки, при этом выполняется операция блокировки до тех пор, пока с сервера не будут получены результаты вывода (синхронный блокирующий вызов). Желаемый подкласс интерфейса модели для использования выбирается пользователем с помощью файла конфигурации **FHiCL**. На сервере используются процессоры **Nvidia T4**, нацеленные на ускорение центра обработки данных.

Преимущества описанного подхода:

- 1) Возможность гибкого распределения ресурсов и потенциальное снижение затрат в зависимости от спроса и задач.
- 2) Этот подход позволяет при обработке событий использовать несколько типов вычислительного оборудования в одной и той же задаче.



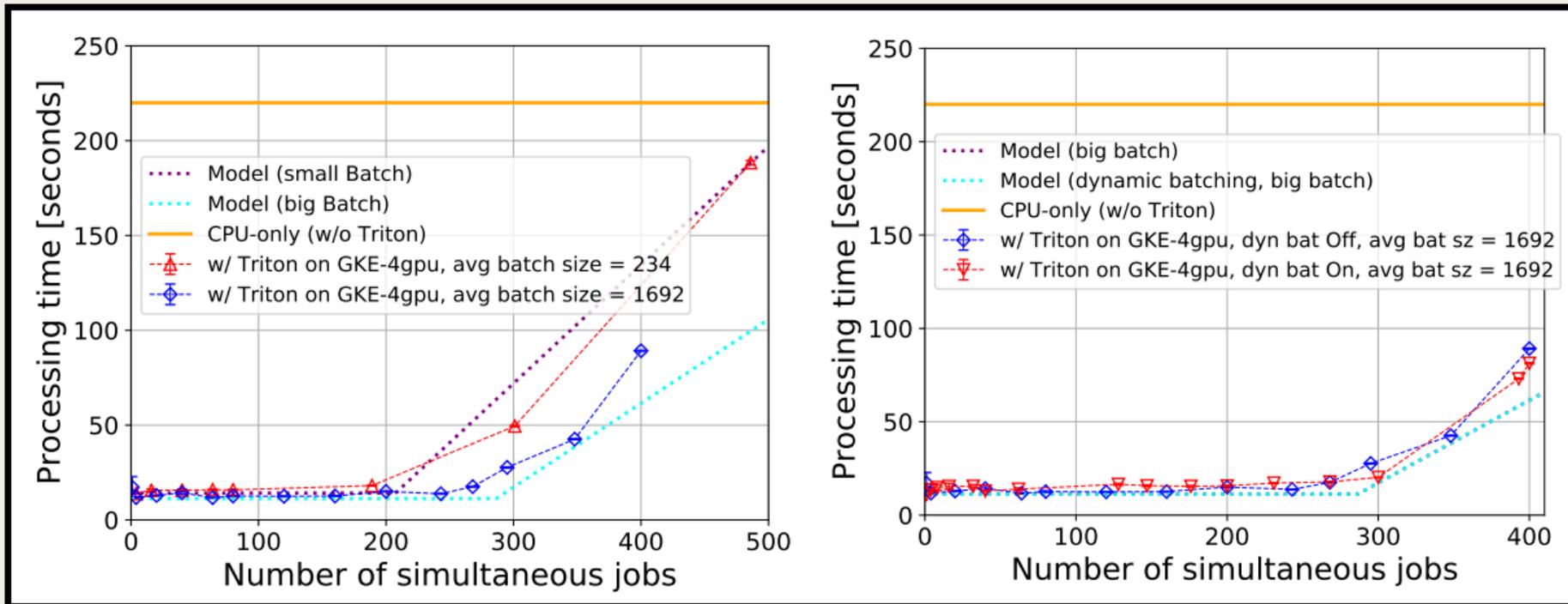
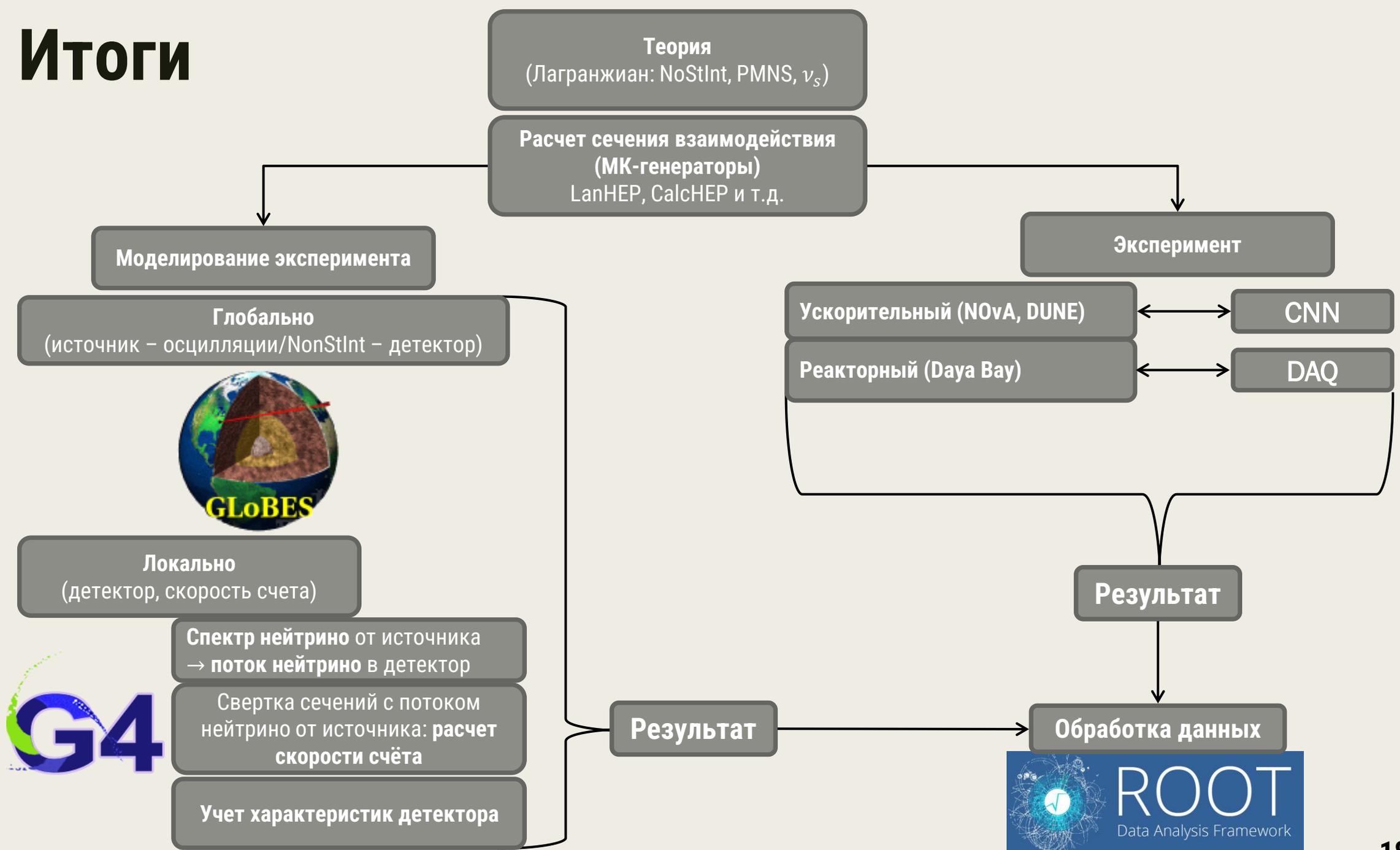


Рис. 17

Время обработки данных модулем EmTrackMichellD как функция числа одновременных процессов CPU. Слева: малый и большой объем данных, с отключенным динамическим пакетированием; Справа: большой объем данных с включенным и выключенным динамическим пакетированием.

Динамическое пакетирование – особенность Nvidia Triton – объединение несколько запросов в пакеты оптимального размера для повышения максимальной эффективности - одновременная обработка нескольких событий без каких-либо изменений в программной структуре эксперимента

Итоги



Список использованных источников

- ***Simulation of long-baseline neutrino oscillation experiments with GLoBES*** / P. Huber [et al.]
Computer Physics Communications (may 2005), vol.167, pp.195-202, [DOI:10.1016/j.cpc.2005.01.003](https://doi.org/10.1016/j.cpc.2005.01.003)
GLoBES: General Long Baseline Experiment Simulator, <https://www.mpi-hd.mpg.de/personalhomes/globes/>
- ***Onsite data processing and monitoring for the Daya Bay Experiment*** / Liu Ying-Biao [et al.]
Chinese Physics C (2014), vol. 38, p. 086001, [DOI:10.1088/1674-1137/38/8/086001](https://doi.org/10.1088/1674-1137/38/8/086001)
- ***Computational techniques for the analysis of small signals in high-statistics neutrino oscillation experiments*** / M.G. Aartsen [et al.]
Nuclear Instruments and Methods in Physics Research Section A (oct. 2020), vol.977, p.164332, [DOI:10.1016/j.nima.2020.164332](https://doi.org/10.1016/j.nima.2020.164332)
- ***Computing for the DUNE Long-Baseline Neutrino Oscillation Experiment*** / H. Schellman [et al.]
EPJ Web Conf. (nov. 2020), vol.245, p.11002, [DOI:10.1051/epjconf/202024511002](https://doi.org/10.1051/epjconf/202024511002)
- ***GPU-accelerated machine learning inference as a service for computing in neutrino experiments*** / M. Wang [et al.]
Frontiers Big Data (jan. 2021), vol.3, [DOI:10.3389/fdata.2020.604083](https://doi.org/10.3389/fdata.2020.604083)

- ***Expanding IceCube GPU computing into the Clouds*** / Igor Sfiligoi [et al.]
IEEE 17th International Conference on eScience (2021), pp. 227-228, [DOI:10.1109/eScience51609.2021.00034](https://doi.org/10.1109/eScience51609.2021.00034)
- ***IceCube Computing Grid*** (запись онлайн-митинга команды IceCube, 2020), [youtube](#)

Спасибо за внимание!