

Быстрая реконструкция траекторий заряженных частиц в эксперименте CBM на основе фильтра Калмана с использованием параллельных вычислений на многоядерном сервере ЛИТ ОИЯИ

Т.О. Аблязимов¹, М.В. Зызак^{2,3}, В.В. Иванов¹, П.И. Кисель¹

¹ Лаборатория информационных технологий, ОИЯИ, Дубна, Россия

² Франкфуртский университет имени Гете, Франкфурт-на-Майне, Германия

³ Киевский национальный университет имени Тараса Шевченко, Киев, Украина

Задача нахождения параметров траекторий заряженных частиц – одна из основных задач в эксперименте CBM (GSI, Германия). Эксперимент предполагает полную реконструкцию событий в режиме реального времени, что требует развития быстрых алгоритмов, максимально использующих потенциал современных архитектур CPU и GPU. В настоящей работе приведены результаты анализа алгоритма реконструкции траекторий заряженных частиц на основе фильтра Калмана, реализованного с применением различных методов распараллеливания кода. Для работы использовался многоядерный сервер ЛИТ ОИЯИ с двумя CPU Intel Xeon X5660 и GPU NVidia GTX 480.

Введение

В настоящее время в GSI (Дармштадт, Германия) на строящемся ускорительном комплексе FAIR (Facility for Antiproton and Ion Research) ведутся работы по созданию экспериментальной установки CBM (Compressed Baryonic Matter). Физическая программа CBM нацелена на всестороннее изучение фазовой диаграммы сильновзаимодействующей материи и уравнения состояния вещества при экстремально высоких плотностях барионной материи [1]. Для ее реализации необходимы измерения распределений множественности в фазовом пространстве и потоков всех интересных с точки зрения физической программы частиц.

Быстрая реконструкция траекторий заряженных частиц с помощью детектора STS (Silicon Tracking System) является ключевой проблемой в задаче отбора полезных событий в эксперименте CBM. Высокая множественность событий (до 1000 треков в каждом ядро-ядерном соударении), интенсивный фон (до 85% фоновых отсчетов STS детекторе), неоднородное магнитное поле и необходимость реконструкции всех событий в режиме реального времени (до 10^7 событий в 1 секунду) требуют не только развития новых подходов для решения рассматриваемой задачи, но и мак-

симального использования потенциала современных многоядерных CPU/GPU архитектур.

Процедура реконструкции траекторий заряженных частиц, регистрируемых с помощью вершинного детектора CBM, включает два последовательных этапа:

- 1) распознавание треков в условиях высокой множественности и плотности заряженных частиц, интенсивного фона и неоднородного магнитного поля,
- 2) восстановление параметров трека (место попадания в координатный детектор и направление трека) и импульса заряженной частицы.

В настоящей работе приведены результаты исследования производительности алгоритма реконструкции траекторий заряженных частиц, реализованного на основе фильтра Калмана. Сервер был оснащен двумя CPU Intel Xeon X5660 и GPU NVidia GTX 480. Для оценки алгоритма использовались различные подходы для распараллеливания и векторизации программного кода: заголовочные файлы, средства библиотеки Vc (Vector Classes) [2], программные среды OpenMP (Open Multi-Processing) [3] и OpenCL (Open Computing Language) [4].

Требования, предъявляемые к алгоритму

К алгоритму реконструкции параметров треков предъявляются два основных требования:

- 1) максимально возможная точность восстановления (x, y) -координат места попадания трека в конкретный координатный детектор, наклонов трека в этой точке и импульса частицы;
- 2) высокая скорость реконструкции треков в реальном времени эксперимента.

Первое играет ключевую роль, в частности, при реконструкции типа и места распада исследуемых в эксперименте наблюдаемых. В свою

очередь, скорость выполнения алгоритма крайне важна для СВМ, поскольку планируется проводить реконструкцию событий в реальном времени эксперимента.

Точность восстановления треков методом фильтра Калмана

Для оценки точности реконструкции параметров трека используем такое понятие, как остаток ρ . При этом остаток ρ_x , в частности, для x -координаты трека в одном из детекторов STS-системы определяется как разность между величиной x_{mc} , полученной в результате Монте-Карло моделирования прохождения трека через детектор, и значением x_{reco} , реконструированным с помощью рассматриваемого алгоритма:

$$\rho_x = x_{reco} - x_{mc}. \quad (1)$$

В качестве характеристики надежности реконструкции параметров трека используются нормированные остатки (пулы):

$$P(x) = \frac{\rho_x}{\sqrt{C_{xx}}}, \quad (2)$$

где C_{xx} – диагональный элемент, соответствующей ковариационной матрицы, полученной в результате реконструкции трека. В идеальном случае, пулы должны быть распределены по закону Гаусса с единичной дисперсией.

Для оценки качества реконструкции треков с помощью рассматриваемого алгоритма нами использовались треки, найденные в STS-системе эксперимента СВМ методом клеточного автомата [5, 6]. Для этого было отобрано 20000 “длинных” первичных треков, координаты которых были зарегистрированы во всех STS-станциях. Качество реконструкции треков оказалось одинаковым для всех реализаций упомянутых выше средств векторизации и распараллеливания кода.

Результаты тестирования представлены на рис. 1. Поскольку распределения для x - и y -координат и соответствующих им наклонам абсолютно идентичны, то результаты приведены только для переменной x .

Распределения пулов хорошо аппроксимируются нормальным законом с дисперсией, близкой к единице. Этот результат указывает на корректность процедуры фитирования.

Масштабируемость алгоритма на центральных процессорах

Для оценки производительности алгоритма использовался многоядерный сервер `cuda.jinr.ru` ЛИТ ОИЯИ, оснащенный двумя процессорами

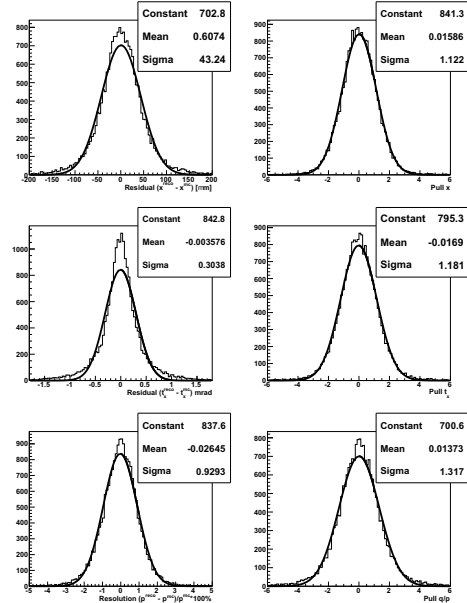


Рис. 1: Распределение остатков и пулов для параметров треков, реконструированных с помощью фильтра Калмана

Intel Xeon X5660, каждый из которых содержит 6 физических ядер с частотой 2.8 ГГц. Используя технологию гиперпоточности одновременно может быть запущено 24 потока.

Для распараллеливания алгоритма на ядра CPU использовались две программные среды: OpenMP и OpenCL.

На рис. 2 приведены графики масштабируемости алгоритма реконструкции треков заряженных частиц в зависимости от числа запущенных в среде OpenMP логических ядер при использовании для векторизации кода заголовочных файлов (слева) и библиотеки Vc (справа).

Оба метода показали одинаковую производительность, позволив достичь скорости обработки 34 трека/мкс.

Использование OpenCL для CPU (рис. 3) также позволило добиться линейной масштабируемости. Результат немного скромнее (27 треков/мкс), а иной характер зависимости объясняется другим порядком подключения логических ядер нежели в случае с OpenMP.

Производительность алгоритма на графических ускорителях и распределение задач по рабочим группам

Реализация алгоритма в среде OpenCL позволила запускать его также и на графических картах. Для тестирования использовалась Nvidia GTX 480. Процессор этой карты содержит 448 ядер CUDA.

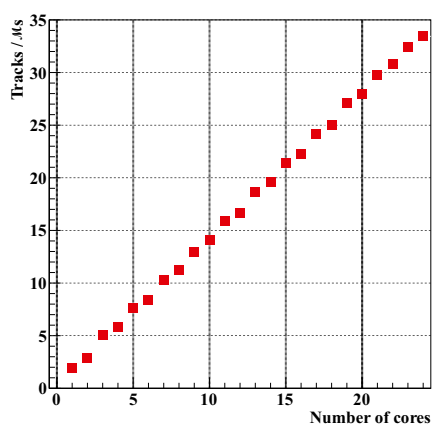
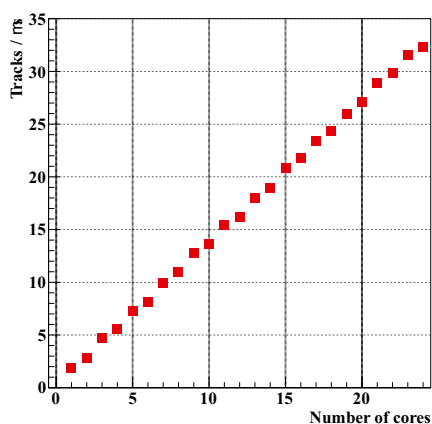


Рис. 2: Масштабируемость алгоритма реконструкции треков заряженных частиц в зависимости от числа запущенных в среде OpenMP логических ядер при использовании для векторизации кода заголовочных файлов (сверху) и библиотеки Vc (снизу)

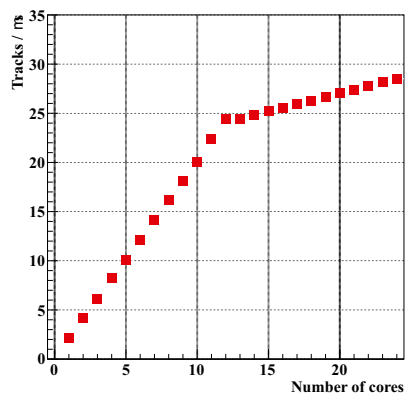


Рис. 3: Масштабируемость алгоритма реконструкции параметров треков заряженных частиц по отношению к числу логических ядер центрального процессора в среде OpenCL

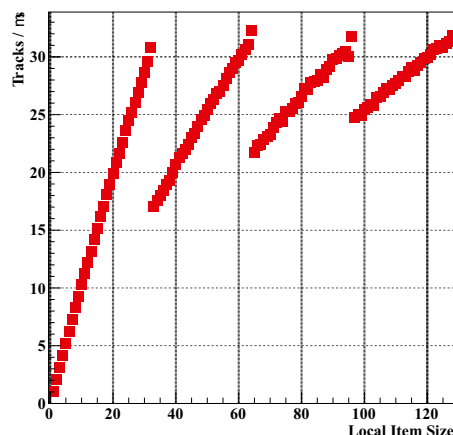


Рис. 4: Производительность графического процессора NVidia GTX 480 в зависимости от числа треков в рабочей группе

При запуске программы весь набор треков, подлежащих обработке, распределяется между рабочими группами. Каждая из них обрабатывается потоковым мультипроцессором (по 32 ядра в каждом).

На рис. 4 можно видеть, что максимальная скорость обработки достигается, если число треков в рабочей группе кратно числу ядер в потоковом мультипроцессоре и составляет 33 трека/мкс.

Заключение

В настоящей работе изучена и продемонстрирована возможность проведения быстрой реконструкции параметров траекторий заряженных частиц, регистрируемых системой координатных детекторов эксперимента СВМ, на основе фильтра Калмана с использованием параллельных вычислений на многоядерном сервере ЛИТ ОИ-ЯИ. К разрабатываемому алгоритму предъявлялись два основных требования: 1) высокая производительность алгоритма, 2) максимально возможная точность реконструкции нужных параметров трека (координата и направление) и частицы (импульс).

Первое требование было выполнено за счет разработки параллельного алгоритма с использованием различных современных средств по распараллеливанию и векторизации кода, таких как заголовочные файлы, библиотека Vc, программные среды OpenMP и OpenCL. Указанные технологии позволяют запускать созданный программный код на высокопроизводительных многоядерных и гибридных системах, оснащенных векторными (SIMD) модулями и графическими ускорителями.

Решение по второму требованию было найдено в результате правильного выбора в качестве метода для реализации указанной задачи рекурсивного фильтра Калмана, а также за счет применения ряда приближений, позволивших без потери точности вычислений обеспечить высокую надежность и скорость обработки.

Разработанный алгоритм был протестирован на сервере `cuda.jinr.ru` ЛИТ ОИЯИ с двумя процессорами Intel Xeon X5660 и графической картой NVidia GTX 480. Все программные реализации показали линейную масштабируемость и высокую производительность (34 трека за 1 мкс на центральном процессоре и 33 трека за 1 мкс на графическом). Таким образом, уже в существующей комплектации сервер позволяет обрабатывать до 70-ти треков за 1 мкс.

Список литературы

- [1] The CBM Physics Book, editing by B. Friman, C. Höhne, S. Leupold, J. Knoll, J. Randrup, R. Rapp, P. Senger, Lecture Notes in Physics, Vol. 814, 1st Edition., 2011, 960 p.
- [2] Vector Classes, (<http://gitorious.org/vc>).
- [3] OpenMP, (<http://openmp.org>).
- [4] OpenCL, (<http://www.khronos.org/opencv1>).
- [5] I. Kisel, Event reconstruction in the CBM experiment. Nucl. Instr. and Meth. A566 (2006) 85-88.
- [6] I.S. Kulakov, S.A. Baginyan, V.V. Ivanov, and P.I. Kisel: Performance Analysis of Cellular Automaton Algorithm to Solve the Track-Reconstruction Problem on a Multicore Server at the Laboratory of Information Technologies, Joint Institute of Nuclear Research, ISSN 1547-4771, Physics of Particles and Nuclei Letters, 2013, Vol. 10, No. 2, pp. 162-170. Pleiades Publishing, Ltd., 2013.