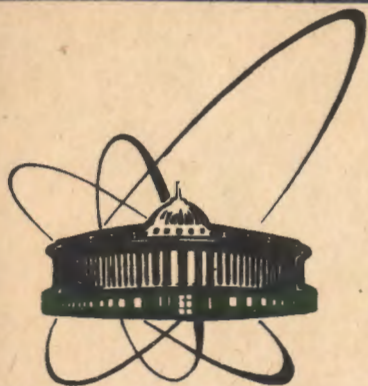


90-504



**СООБЩЕНИЯ
ОБЪЕДИНЕННОГО
ИНСТИТУТА
ЯДЕРНЫХ
ИССЛЕДОВАНИЙ
ДУБНА**

P10-90-504

**А.М.Задорожный, И.М.Иванченко, В.В.Пальчик,
В.В.Старков**

**ПРИМЕНЕНИЕ ПАРАЛЛЕЛЬНЫХ ВЫЧИСЛЕНИЙ
ДЛЯ ОБРАБОТКИ ЭКСПЕРИМЕНТАЛЬНЫХ ДАННЫХ
ФИЗИКИ ВЫСОКИХ ЭНЕРГИЙ**

1990

Введение

В настоящее время наиболее перспективный путь повышения производительности обработки данных - использование параллельных вычислительных систем ^{1/}.

Один из эффективных подходов к решению проблемы параллельной обработки данных ускорительных экспериментов реализован на базе мультипроцессорных систем ^{2,3/} с простой классической топологией. В такого типа вычислительных комплексах (фермах) реализован принцип организации однородной обработки во всех процессорах, при этом существенно используется естественный параллелизм данных: узловые процессоры (микро-ЭВМ, эмуляторы, транспьютеры) обрабатывают элементарные единицы данных - события целиком. При несколько иных свойствах вычислительной среды (например, универсальные мультипроцессорные векторно-конвейерные ЭВМ) ^{4/} возникает необходимость сочетания нескольких типов параллелизма.

Одним из препятствий в применении параллельных систем является низкая эффективность выполнения такими системами традиционных скалярных программ. Скалярную программу необходимо предварительно преобразовать, чтобы сделать возможным одновременное (параллельное) выполнение операций, решение различных задач или частей одной задачи. Практически все программы в большей или меньшей степени допускают распараллеливание. К сожалению, эффективно автоматизировать этот процесс пока не удастся, и в каждом случае требуются существенные усилия разработчиков. Такие усилия направляются в первую очередь на пакеты программ, в которых ожидается наибольший эффект. В физике высоких энергий особый интерес представляет распараллеливание программ моделирования установок и обработки экспериментальных данных.

Целью настоящей работы является изучение возможности и целесообразности использования параллельных вычислительных устройств (в данном случае матричных процессоров ЕС-2706), взаимодействующих со скалярной универсальной ЭВМ, для обработки данных электронных экспериментов, проводимых в ОИЯИ.

При обработке экспериментальных данных значительная часть времени тратится на распознавание траекторий частиц (треков). В связи с этим объектом распараллеливания была выбрана программа LINTR ^{/5/}, которая применяется для поиска прямолинейных треков, регистрируемых бесфильмовыми координатными детекторами. Сравнение производительности векторного и скалярного вариантов программы проводилось на событиях, сгенерированных методом Монте-Карло, а также на реальных событиях (эксперименты на установке БИС-2) ^{/6/}. Результаты, полученные с использованием одного матричного процессора ЕС-2706, обнадеживают: коэффициент ускорения по отношению к ЕС-1061 на реальных событиях от 4 до 7.

План дальнейшего изложения следующий. В первой части обсуждается постановка задачи, выбор объекта распараллеливания, языка программирования. Вторая часть работы посвящена особенностям матричного процессора (МП) ЕС-2706 и способам достижения максимальной производительности МП. В третьей части приведены результаты распараллеливания программы LINTR. Вопросы организации макроконвейера рассматриваются в четвертой части.

I

При выборе объекта распараллеливания мы руководствовались следующими принципами:

- 1) универсальность и гибкость программы, возможность ее применения в различных системах обработки;
- 2) времяемкость, т.е. частое использование этой программы и/или значительные затраты времени на ее выполнение;
- 3) небольшой размер (порядка 1000 фортрановских строк) - разработка программ для МП, как правило, требует значительно больше времени, чем для скалярной ЭВМ;
- 4) время выполнения программы на МП должно превышать время доступа к памяти, чтобы не утратить преимуществ применения МП, используя программу, при выполнении которой большая часть времени расходуется на обмен данными между основной машиной и МП.

Всем вышеперечисленным требованиям в достаточной степени

отвечает программа LINTR, используемая, как отмечалось выше, в системах обработки данных экспериментов физики высоких энергий и предназначенная для распознавания треков в двумерной области. Программа LINTR не является ориентированной на определенную установку и использовалась для обработки многих экспериментов. Особенностью этой программы является высокая эффективность распознавания и удовлетворительное время работы при минимальном объеме памяти.

Существует три языка программирования МП ЕС-2706. Это AP-FORTRAN, VFC и APAL. Охарактеризуем каждый из этих языков программирования, их достоинства и недостатки. Транслятор с языка Фортран генерирует неэффективный код, который не реализует на приемлемом уровне возможность параллельной обработки, присущей МП.

Язык VFC - язык достаточно высокого уровня. Основной структурной единицей языка VFC является предварительно созданная микропрограмма. Язык VFC представляет удобные средства для организации выполнения допускающих внешнюю параметрическую настройку микропрограмм. Одно из главных достоинств использования языка VFC связано с высокой эффективностью организации передачи данных между ЦП и МП.

Для достижения максимального быстродействия самым подходящим языком программирования МП является язык APAL, который позволяет учитывать все особенности архитектуры МП ЕС-2706 (архитектуру МП мы рассмотрим в следующей части статьи).

К недостаткам языка APAL можно отнести то, что время, необходимое для написания и отладки программы на APAL, в несколько раз превосходит время для написания и отладки программы на Фортране. Несмотря на это, для реализации максимального быстродействия, мы остановились при распараллеливании LINTR на языке APAL.

II.

МП ЕС-2706 - это периферийный процессор, работающий параллельно с центральным процессором (ЦП). Характерными особенностями архитектуры МП являются наличие в его составе

нескольких субпроцессоров и конвейеризация ряда операционных элементов. На рис.1 представлена структурная схема МП.

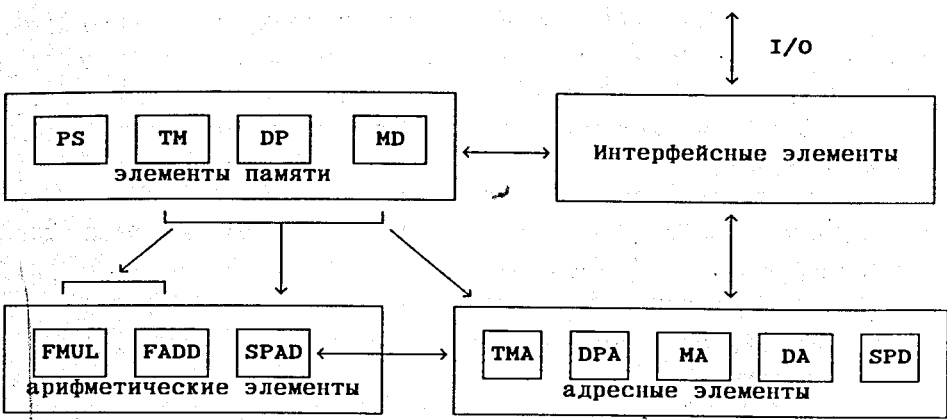


Рис.1.
Структурная схема МП ЕС-2706.

Устройства МП могут быть разделены на запоминающие, арифметические, адресные, интерфейсные и управляющие. Мы рассмотрим только запоминающие и арифметические устройства в объеме, необходимом разработчикам программного обеспечения.

МП ЕС-2706 имеет четыре типа запоминающих устройств, которые разделены структурно и функционально: память микропрограмм (PS), память данных (MD), память констант (TM) и буферная память данных (DP). Они связаны с операционными элементами посредством шин и могут работать одновременно.

В состав МП входят три арифметических устройства, которые могут работать одновременно и независимо. Это сумматор (FADD) и умножитель (FMUL) чисел с плавающей запятой и блок целочисленной арифметики (SPAD) (рис.1). Сумматор и умножитель организованы соответственно, как двухступенчатый и трехступенчатый конвейер. На каждом такте МП можно инициировать сумматор и умножитель, а результат выполнения операций можно получить соответственно через 2 и 3 такта МП.

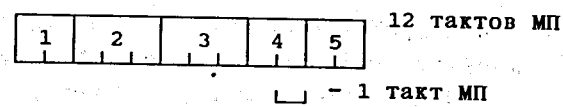
Рассмотрим некоторые способы повышения производительности МП при программировании на АРАЛ. В приведенных ниже примерах

временные характеристики программ даются в числе тактов работы процессора. Каждый такт имеет длительность 1/6 мкс.

Пример 1. Параллельное выполнение операций при вычислении арифметических выражений;

$$\begin{matrix} (a + b) * c + d * e + f \\ 1 \quad 2 \quad 4 \quad 3 \quad 5 \end{matrix} \quad \text{— номер операции}$$

Последовательный вариант

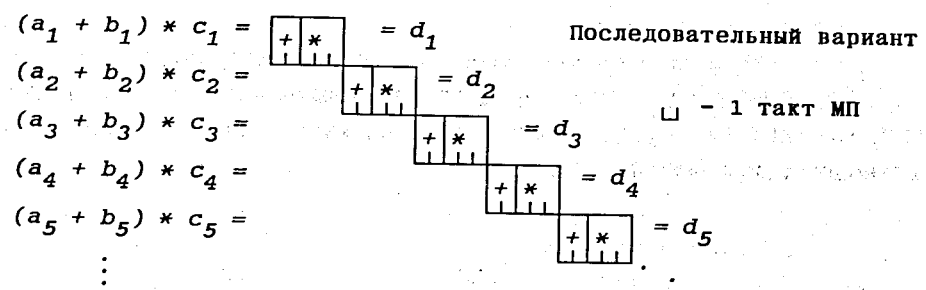


Параллельный вариант

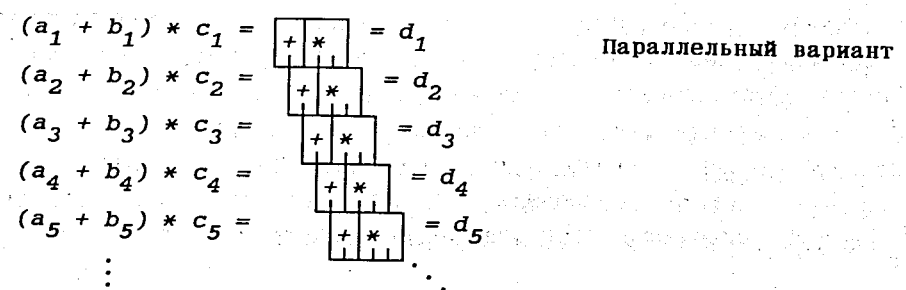


Пример 2. Параллельное выполнение нескольких шагов цикла.

$$(a_i + b_i) * c_i = d_i$$



$a * n$ - время вычисления n-элементов вектора d (a-время вычисления элемента вектора)



$(a-b)+b*n$ - время вычисления n -элементов вектора d (a -время вычисления первого элемента вектора, b -время вычисления каждого следующего элемента вектора).

Коэффициент ускорения $K_n = \frac{a*n}{(a-b)+b*n}$,

$\lim_{n \rightarrow \infty} K_n = \frac{a}{b}$.

В заключение этого раздела перечислим факторы, позволяющие получить высокую производительность матричного процессора :

- небольшое количество тактов, требуемых для выполнения операций (самая длинная операция - умножение с плавающей запятой занимает всего три такта);
- конвейеризация ряда функциональных устройств;
- возможность одновременной работы различных функциональных устройств МП;
- наличие нескольких типов памяти с различными временными характеристиками;
- независимые пути передачи данных (большое количество шин).

При реализации потенциальных возможностей рассматриваемого вычислительного устройства определяющую роль играет его математическое обеспечение.

III

В алгоритме LINTR существенно учитывались такие особенности регистрации событий бесфильмовыми координатными детекторами, как стохастичность регистрации точки (в реальных условиях эффективность детекторов находится в пределах 90-100%), малое количество точек на треке (каждая точка регистрируется отдельной камерой), значительное расстояние между точками (большие расстояния между камерами), прямолинейность распознаваемых траекторий (детекторы установлены вне магнитного поля).

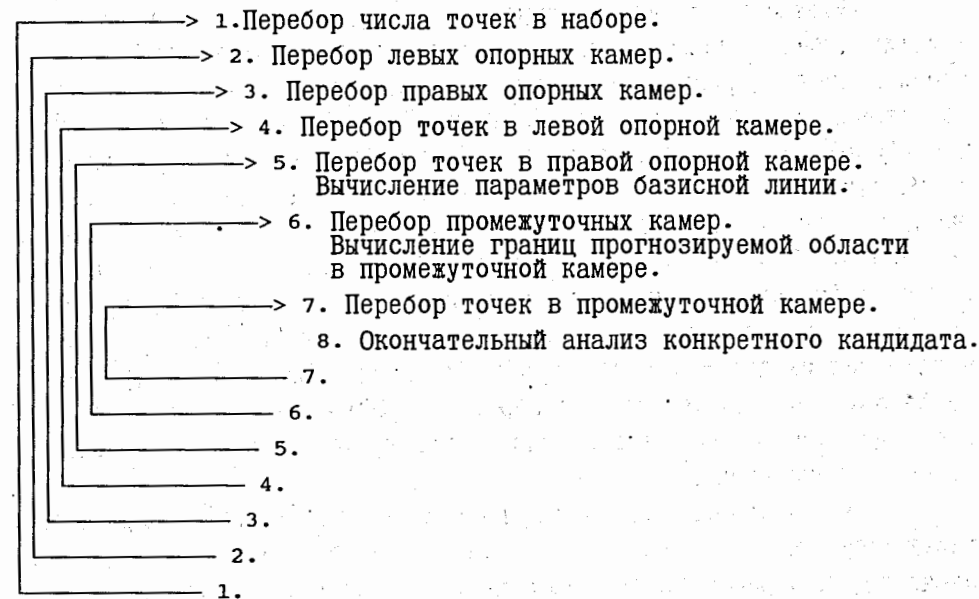


Рис.2 Схема алгоритма LINTR.

Основу алгоритма составляет направленный перебор, реализуемый системой вложенных циклов, и анализ кандидатов в треки (рис. 2). Область поиска определяется окрестностью базисной линии, проведенной через точки в опорных камерах. Множество значений опорных точек выбирается из условия минимума прогнозируемой области в процессе присоединения очередного элемента к искомому набору точек. Опорные точки являются одновременно и граничными точками в наборе.

Формирование набора точек осуществляется для допустимой по углу наклона и проходящей через область мишени базисной прямой. Сокращение времени просмотра промежуточных камер и точек в них достигается благодаря упорядоченности обеих информационных структур и их представлению в матричном виде.

На стадии отбора точек в промежуточных камерах предусмотрено сокращение просмотра точек за счет отбрасывания тех из них, которые анализировались для предыдущей базисной прямой. Эффект этой процедуры растет с увеличением числа камер, треков и фоновых точек в событии.

Таблица 2

NTR+BG	EC2706 (sec./ev.)	EC1061 (sec.ev.)	$\frac{\text{cp TIME EC1061}}{\text{ast TIME EC2706}}$
1+1	0.00067	0.00290	4.3
2+1	0.00094	0.0040	4.3
3+1	0.00122	0.0050	4.1
4+1	0.00148	0.0061	4.1
5+1	0.00182	0.0076	4.2
8+1	0.00291	0.0123	4.2
10+1	0.00373	0.0162	4.3

Таблица 3

NTR+BG	ast TIME EC2706 (sec./ev.)	cp TIME EC1061 (sec./ev.)	$\frac{\text{cp TIME EC1061}}{\text{ast TIME EC2706}}$
1+1	0.00067	0.0029	4.3
2+2	0.00150	0.0073	4.8
3+3	0.0027	0.0146	5.4
4+4	0.0045	0.0253	5.6
5+5	0.0069	0.0400	5.7
8+8	0.0172	0.0965	5.7
10+10	0.0272	0.1510	5.6

Из таблиц 1, 2, 3 видно, что повышение относительной производительности МП к ЦП с ростом числа треков (NTR) меняется незначительно. Это можно объяснить быстрым уменьшением числа переборов вследствие вычеркивания точек для найденных треков. Значительный рост производительности происходит

Варианты рассмотренной программы с небольшими модификациями использовались в ряде экспериментов ОИЯИ. Выбор определенной версии программы распознавания зависит от типа изучаемых событий, фоновых условий, эффективности и точности детекторов. В случае близкорасположенных фоновых точек или высокой плотности искомых траекторий реализуется процедура перебора без выбывания найденных точек: с сохранением отобранных точек в течение одного шага. Среди конкурирующих наборов с одинаковым количеством точек выбираются наилучшие по дополнительному критерию качества. Радикальное уменьшение времени достигается за счет исключения из рассмотрения ложных комбинаций благодаря встроенному механизму использования априорной информации и оптимальному рассредоточению решающего правила - непрерывному контролю развития процесса присоединения точек.

В таблицах 1, 2, 3 приведены сравнительные временные характеристики скалярного (cp TIME) и параллельного (ast TIME) вариантов программы LNTR, полученные на событиях, сгенерированных методом Монте-Карло. Во всех этих случаях количество камер в плоскости XZ равно 5, в плоскости YZ - 6, максимальное количество точек на треке соответственно 5 и 6, минимальное для обеих плоскостей - 4. Характеристики приводятся для различных значений количества треков (NTR) и фоновых точек (BG).

Таблица 1

NTR+BG	ast TIME EC2706 (sec./ev.)	cp TIME EC1061 (sec./ev.)	$\frac{\text{cp TIME EC1061}}{\text{ast TIME EC2706}}$
1+0	0.00036	0.00114	3.2
2+0	0.00053	0.00161	3.0
3+0	0.00073	0.00219	3.0
4+0	0.00095	0.00284	3.0
5+0	0.00120	0.00360	3.0
8+0	0.00200	0.00680	3.3
10+0	0.00280	0.00960	3.5

с ростом числа фоновых точек (BG). На рис.3 представлен график зависимости роста производительности параллельного варианта программы LINTP по отношению к скалярному варианту программы с ростом числа фоновых точек при фиксированном числе треков (NTR=5).

$\frac{\text{cp TIME EC1061}}{\text{ast TIME EC2706}}$

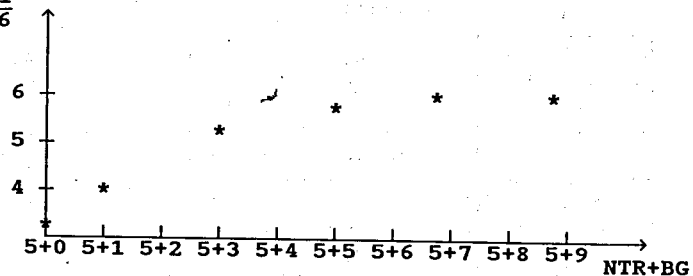


Рис. 3.

Из рис.3 видно, что насыщение конвейера достигается достаточно быстро. Уже при 5 фоновых точках конвейер выходит на плато, и дальнейший рост относительной производительности незначителен.

В таблицах 4 и 5 приведены данные о производительности параллельного и скалярного варианта программы LINTP на реальных событиях (БИС-2) в зависимости от минимального количества точек (MNSPA) на искомом треке. В таблице 4 приведены данные о событиях со средним числом треков 5,7 и со средним числом фоновых точек на камеру 4,2 . В таблице 5 представлены данные о событиях со средним числом треков 11,6 и средним числом фоновых точек на камеру 10,7.

Таблица 4

MNSPA		ast TIME EC2706 (sec./ev.)	cp TIME EC1061 (sec./ev.)	$\frac{\text{cp TIME EC1061}}{\text{ast TIME EC2706}}$
XZ	YZ			
3	3	0.0153	0.0654	4.2
4	4	0.0112	0.0541	4.8
5	5	0.0060	0.0328	5.4
5	6	0.0039	0.0197	5.8

Таблица 5

MNSPA		ast TIME EC2706 (sec./ev.)	cp TIME EC1061 (sec./ev.)	$\frac{\text{cp TIME EC1061}}{\text{ast TIME EC2706}}$
XZ	YZ			
3	3	0.0670	0.294	4.3
4	4	0.0510	0.273	5.4
5	5	0.0261	0.147	5.7
5	6	0.0140	0.109	7.4

Из представленных таблиц видно , как растет относительная производительность параллельной программы с ростом минимальной длины искомого трека и с ростом сложности событий.

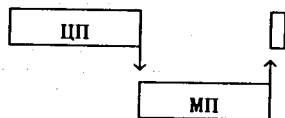
Надо отметить полную идентичность выходных данных параллельного и скалярного варианта, вплоть до совпадения порядка нахождения треков.

IV

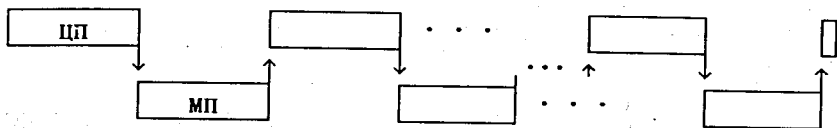
МП ЕС-2706 по отношению к ЦП является внешним вычислительным устройством, что позволяет говорить о распараллеливании не только в рамках МП , но и распараллеливании между МП и ЦП, о так называемом макроконвейере.

При традиционной организации работы- двухмашинного комплекса МП - ЦП основная программа , выполняемая на ЦП , вызывает выполнение подпрограммы или набора подпрограмм, выполняемых на МП. Подпрограммы , выполняемые на МП , вызываются с помощью обычного оператора вызова подпрограмм (CALL). После выполнения оператора CALL происходит переход к системной программе , которая пересылает в МП подпрограмму и данные и инициирует работу МП. Программа ЦП переходит в состояние ожидания, в котором она находится до тех пор, пока МП не выполнит вычисления. По окончании работы МП системная программа обеспечивает пересылку полученных результатов в ЦП и возврат к основной программе.

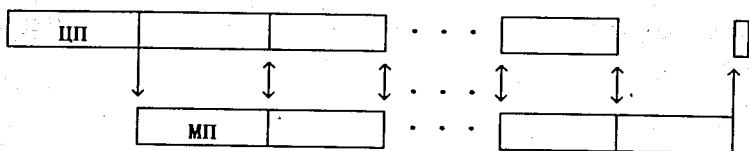
Такую организацию работы комплекса ЦП - МП схематично можно представить в виде :



При неоднократном вызове подпрограмм последовательность работы процессоров выглядит так :



Мы реализовали другую схему - макроконвейер - с одновременным использованием обоих процессоров :



Для сокращения числа обменов ЦП ↔ МП нами была реализована такая архитектура программы, которая позволила передавать исходные данные в МП и полученный результат в ЦП не для одного события, а для группы событий, занимающих 20 Кслов МП (1 слово МП = 38 бит). Для экспериментальных данных (БИС-2) в такую память помещается порядка 180 событий. Тем самым мы сократили число обменов ЦП ↔ МП до уровня, при котором время обменов пренебрежимо мало по сравнению со временем выполнения программы на МП.

Заключение

Полученные результаты показывают возможность эффективного использования МП ЕС-2706 в программах обработки экспериментальных данных в физике высоких энергий со значительным ускорением обработки. Разработанная архитектура программы позволяет

одновременно использовать центральную машину и внешние вычислительные устройства для геометрической реконструкции событий.

Авторы выражают благодарность Н.С.Ангелову и И.В.Пузынину за полезные обсуждения и поддержку.

Литература

1. Ларионов А.М., Майоров С.А., Новиков Г.И. *Вычислительные комплексы, системы и сети*. Ленинград, Энергоатомиздат, 1987.
2. Gaines I. et al. *The ACP Multiprocessor system at Fermilab*. CPC 1987, v.45, p.323.
3. Booth S.P. et al. *Large Scale Applications of Transputers in HEP: the Edinburgh Concurrent Supercomputer Project*. Edinburgh Preprint 89/463, Edinburgh, 1989.
4. Doran R.W. *Programming for Multiprocessing. (On the Cray X-MD)*. CERN, DD/89/1, Geneva, 1989.
5. Говорун Н.Н. и др. *Программы распознавания прямолинейных треков, регистрируемых посредством искровых камер*. ОИЯИ, 10-7303, Дубна, 1973.
6. Алеев А.Н. и др. *БИС-2 - спектрометр для поиска и исследования узких резонансов*. ОИЯИ, P1-89-854, Дубна, 1989.

Рукопись поступила в издательский отдел
2 ноября 1990 года.