УДК 004.42
https://doi.org/10.47533/2020.1606-146X.216

Е. Г. КЛЮЕВА

Карагандинский технический университет,
Караганда, Казахстан
E-mail: e.klyueva@kstu.kz

РАЗРАБОТКА ПРИЛОЖЕНИЯ ДЛЯ ОПРЕДЕЛЕНИЯ ОПТИМАЛЬНОЙ ФОРМЫ РАЗБИЕНИЯ ЭЛЕМЕНТОВ МАТРИЦ БОЛЬШОЙ РАЗМЕРНОСТИ ДЛЯ УМНОЖЕНИЯ НА ТРЕХ ГЕТЕРОГЕННЫХ ПРОЦЕССОРАХ

В статье представлены результаты разработки web-приложения для определения оптимальной формы разбиения элементов больших матриц между тремя абстрактными гетерогенными процессорами при их перемножении. Также представлены пять классов алгоритмов параллельного умножения матриц: последовательная коммуникация с барьером (Serial Communication with Barrier, SCB), параллельная коммуникация с барьером (Parallel Communication with Barrier, PCB), последовательная коммуникация с перекрытием (Serial Communication with Bulk Overlap, SCO), параллельная коммуникация с перекрытием (Parallel Communication with Bulk Overlap, PCO), параллельное перекрытие с чередованием (Parallel Interleaving Overlap, PIO). Для определения коммуникационной сложности рассматриваемых алгоритмов была использована модель Хокни. В исследовании используются шесть непрямоугольных форм разбиения элементов (Square Corner, Rectangle Corner, Square Rectangle, Block Rectangle, L-Rectangle, Traditional 1D Rectangular), выявленных в исследовании Эшли Дэ Флюмьер [1] в ходе применения технологии «push» для перераспределения элементов матрицы между процессорами. Для определения оптимальности формы при определенных технических условиях используются математические модели, описанные в работах [2, 3]. Для реализации web-приложения были выбраны языки программирования Python и JavaScript, менеджер пакетов pip и технология Ajax. Решение проблемы оптимального разбиения элементов матрицы между процессорами позволит эффективно распределять вычислительные ресурсы для решения прикладных задач в различных научных областях, использующих умножение матриц большой размерности.

Ключевые слова*: модель Хокни, умножение матриц, параллельное программирование, разбиение данных, гетерогенные вычислительные системы, web-приложение.*

Введение. Современные научные и прикладные области исследований решают задачи, использующие большие массивы данных. Для эффективности вычислений используются параллельные вычислительные системы различной сложности и комплектации. С начала 1990-х годов существует необходимость формирования сравнительной характеристики метрик суперкомпьютеров. Для реализации данной задачи был создан список Top500 [4], представляющий собой проект для описания и сравнения 500 наиболее мощных и общественно значимых вычислительных систем в мире. Список обновляется каждые шесть месяцев, что позволяет ему содержать актуальный перечень самых быстрых суперкомпьютеров. Анализ списка за последние несколько лет показывает все нарастающую тенденцию использования в вычислительных структурах гетерогенных компонентов, в числе которых многоядерные и графические процессоры (GPU).

Матричные операции линейной алгебры находят широкое применение в разнообразных научных исследованиях, в том числе в области параллельного программиро-

вания и высокопроизводительных вычислений. Умножение матриц является одной из наиболее важных операций над матрицами. Оно широко используется в таких областях, как теория сетей, решение линейных систем уравнений, преобразование систем координат, моделирование популяций и многих других.

Разбиение данных является важным аспектом для решения задач линейной алгебры. Оно заключается в определении способа распределения элементов матрицы между доступными вычислительными элементами. Разбиение данных позволяет оптимизировать такие показатели, как время выполнения задачи и энергоэффективность.

Методы и материалы исследования. Целью разбиения данных является оптимальное распределение вычислительной нагрузки между доступными процессорами для умножения матриц.

Рассматриваются гетерогенные вычислительные системы, в которых гетерогенность или неоднородность в вычислениях проявляется тремя способами:

– различные значения вычислительной мощности систем;

– различные значения пропускной способности систем;

– комбинация обоих факторов.

В рамках данной работы были сделаны следующие допущения:

– изучаются большие квадратные матрицы размером N элементов, где матрицы A и B являются исходными, а матрица C – результирующей;

– для выполнения умножения применяются три абстрактных процессора P, R и S с различными вычислительными мощностями, определенными отношением $P_r : R_r : S_r$, где P – самый мощный процессор, а $S_r = 1$. Общая вычислительная мощность системы равна $T = P_r + R_r + S_r$. Каждый абстрактный процессор может быть представлен при проведении эксперимента группой процессоров или кластеров, т.к. в данном случае прогнозируемая производительность соответствует экспериментальной [5, 6];

– распределение элементов между процессорами осуществляется в соответствии с их мощностями;

– процессоры соединены полносвязной топологией, где β_1 – латентность среды передачи между процессорами P и S, β_2 – между процессорами P и R, β_3 – между процессорами S и R соответственно;

– в исследовании анализируются шесть потенциальных форм разбиения элементов, представленных на рисунке 1: (Square Corner (SC), Rectangle Corner (RC), Square Rectangle (SR), Block Rectangle (BR), L-Rectangle (LR), Traditional 1D Rectangular (TR)) [1];

– математические модели коммуникационной трудоемкости рассматриваемых алгоритмов построены на основе модели Хокни.

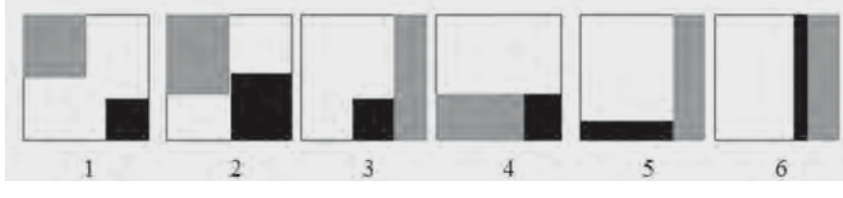

Рисунок 1 – Формы-кандидаты:
1) Square Corner; 2) Rectangle Corner; 3) Square Rectangle;
4) Block Rectangle; 5) L-Rectangle; 6) Traditional 1D Rectangular

При перемножении матриц на нескольких процессорах могут быть использованы алгоритмы следующих классов:

– последовательная коммуникация с барьером (Serial Communication with Barrier, SCB);

– параллельная коммуникация с барьером (Parallel Communication with Barrier, PCB);

– последовательная коммуникация с перекрытием (Serial Communication with Bulk Overlap, SCO);

– параллельная коммуникация с перекрытием (Parallel Communication with Bulk Overlap, PCO);

– параллельное перекрытие с чередованием (Parallel Interleaving Overlap, PIO) [1].

Оптимальность форм-кандидатов оценивалась для каждого из пяти классов алгоритмов.

Первые два алгоритма SCB и PCB базируются на идее массовой коммуникации с барьером, при которой все данные отправляются процессорами до начала вычислений (рисунок 2).

Рисунок 2 – Алгоритмы SCB и PCB

Алгоритмы SCO и PCO предусматривают одновременное выполнение коммуникации и вычислений с целью минимизации затрачиваемого времени (рисунок 3). Алгоритм PIO предполагает, что на каждом шаге часть данных отправляется соответствующим процессором всем остальным вычислительной системы процессорам, которым требуются текущие элементы, в то время как параллельно происходит вычисление результирующей матрицы с использованием данных, которые уже были отправлены. Рассмотренные алгоритмы подробно описаны в работе [1].

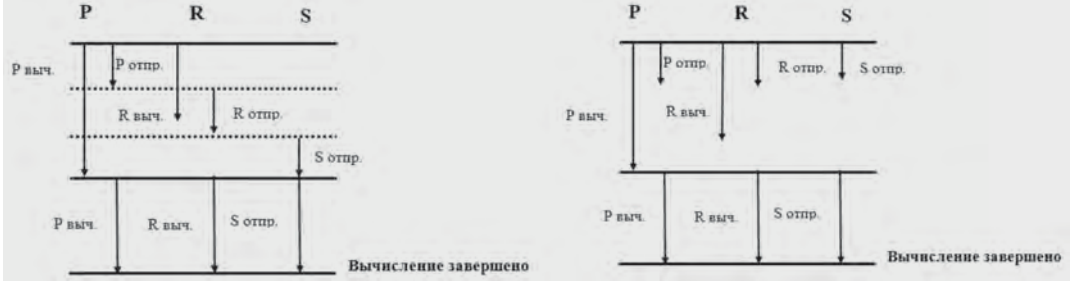

Рисунок 3 – Алгоритмы SCO и PCO

Для определения оптимальности форм для каждого из алгоритмов параллельного умножения матриц при разработке web-приложения были использованы математические модели, описанные в работах [2, 3].

Для реализации web-приложения были приняты следующие решения:

– в качестве языка программирования для написания web-приложения был выбран язык Python;

– в качестве фреймворка использован фреймворк Django, обладающий высокой функциональностью и большим набором готовых решений, способствующих упрощению процесса написания кода;

– для установки фреймворка Django и необходимых библиотек Python был использован менеджер пакетов pip;

– шаблоны, необходимые для отображения страниц web-приложения, представляют собой html-страницы с использованием стилей языка программирования и разметки CSS;

– для разработки функциональности web-приложения на стороне клиента был выбран язык программирования JavaScript;

– для загрузки данных на странице без ее обновления была выбрана технология Ajax;

– для осуществления контроля версий была выбрана система Git.

Результаты и их обсуждение. Функциональность разработанного приложения включает в себя:

– расчет времени вычисления параллельного умножения матриц большой размерности на основе математических моделей для алгоритмов SCB, PCB, SCO, PCO, PIO в соответствии с введенными исходными параметрами вычислительной системы;

– отображение интерактивных графиков времени выполнения для каждого алгоритма по формам разбиения элементов матрицы SC, BR, SR, LR и RC.

Структура разработанного web-приложения приведена на UML-диаграмме классов, изображенной на рисунке 4.

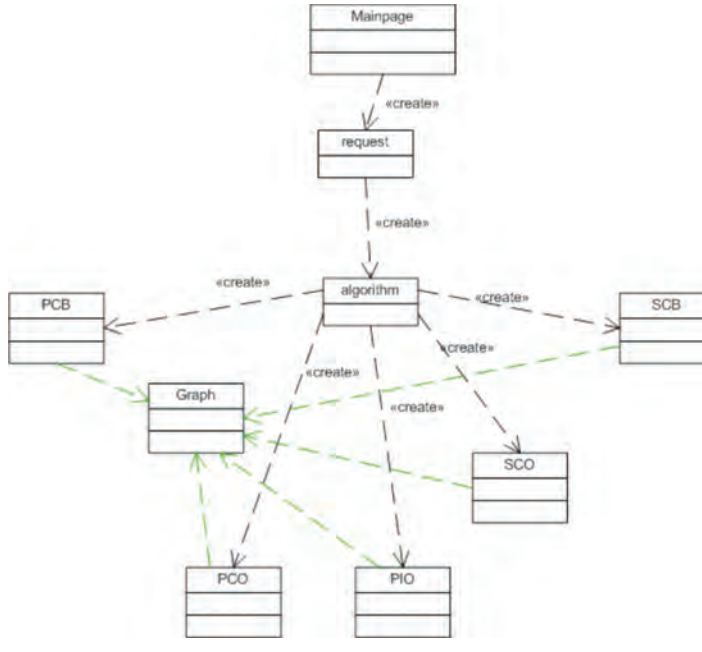

Рисунок 4 – UML-диаграмма классов web-приложения «Heterogeneous Computing»

Web-приложение представляет собой одностраничный сайт. Для отображения графиков применяется функция Ajax, которая подгружает данные на веб-странице.

Пользователю предоставляется возможность ввода следующих переменных:

– P – вычислительная мощность процессора P;
– R – вычислительная мощность процессора R;
– b1 – отношение пропускных способностей между процессорами β1/β3;
– b2 – отношение пропускных способностей между процессорами β2/β3;
– Sp – количество операций, выполняемых процессором P в секунду;
– N – размерность матрицы.

Начальный вид главной страницы web-приложения показан на рисунке 5. Кнопки SCB, PCB, PCO, SCO, PIO предназначены для получения расчетов по соответствующему алгоритму.

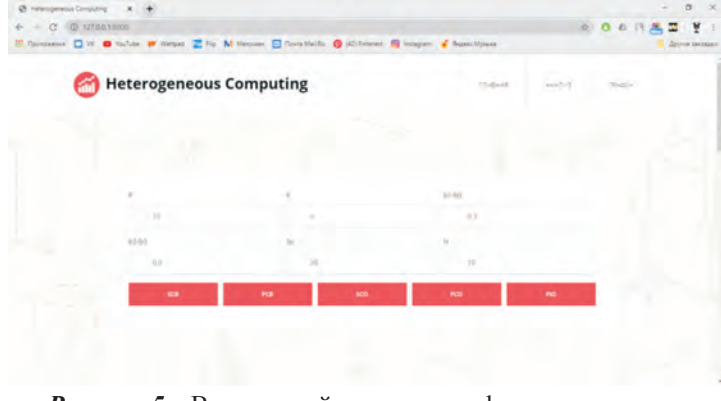

Рисунок 5 – Вид главной страницы web-приложения

Ниже начинается новый блок web-страницы, он является динамичным и изменяется в соответствии с введенными пользователем параметрами благодаря функции Ajax. Заголовок блока представляет собой название выбранного алгоритма. В левой части экрана отображается график и слайдеры для изменения значений параметров P и R. Таким образом, изменяя положение слайдера, можно наглядно увидеть зависимость расположения плоскостей на графике от значений P и R. В правой части экрана приведены расчеты для текущего алгоритма и каждой из пяти форм разбиения. Внизу приведена оптимальная форма разбиения для данного алгоритма и показано ее схематическое изображение.

Соответствующий алгоритмам SCB, PCB, SCO, PCO, PIO анализ для исходных параметров P = 10, R = 4, $\beta_1/\beta_3 = 0{,}3$, $\beta_2/\beta_3 = 0{,}9$ приведен на рисунках 6 – 8.

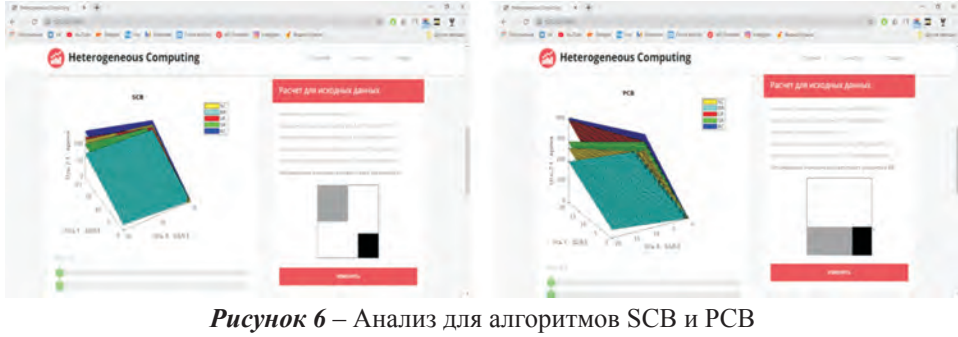

Рисунок 6 – Анализ для алгоритмов SCB и PCB

Рисунок 7 – Анализ для алгоритма SCO и PCO

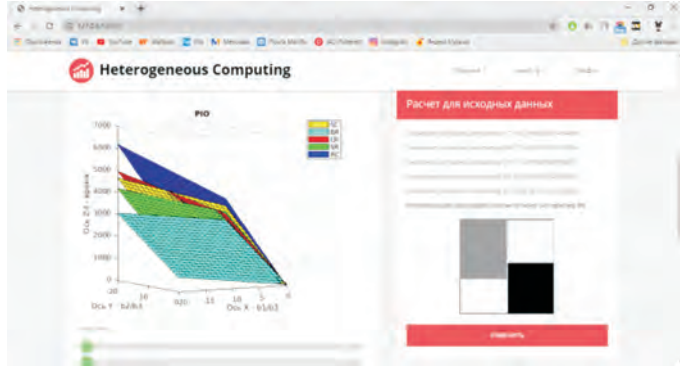

Рисунок 8 – Анализ для алгоритма PIO

В следующем блоке размещен трехмерный график для каждого из классов алгоритмов. Он является динамическим и позволяет рассмотреть плоскости с разных ракурсов, сохранить результаты в формате png, просмотреть соответствующие значения параметров. Данный блок показан на рисунке 9.

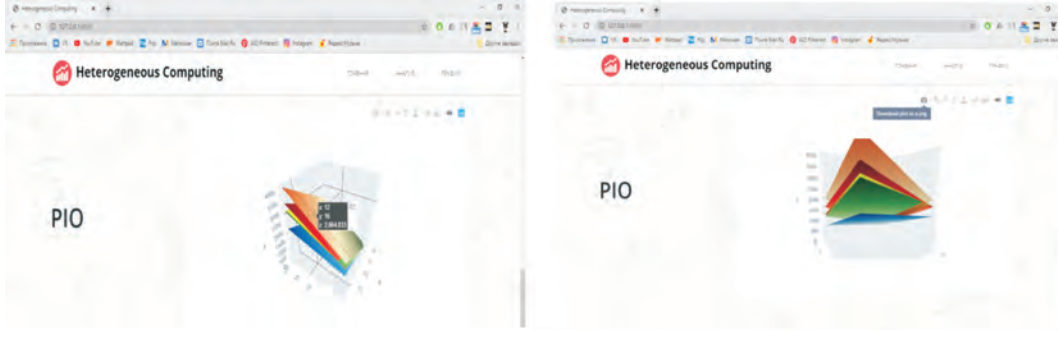

Рисунок 9 – Трехмерный график для алгоритма PIO

В ходе исследования при помощи разработанного web-приложения был проведен анализ различных входных значений. Полученные данные подтвердили выводы, сделанные в работе [3]. Формы разбиения BR, LR, SC и SR являются оптимальными

для алгоритма последовательной коммуникации с барьером (SCB). Форма разбиения элементов LR оптимальна при $\beta_1/\beta_3 < 1$, если мощности процессоров P и R приблизительно равны и значительно больше, чем у процессора S. Форма разбиения SR оптимальна при малых значениях соотношений β_1/β_3, β_2/β_3 и мощностях процессоров P и R, значительно больших чем у S. Формы BR и SC оптимальны в остальных случаях, при этом выбор зависит от конкретных значений исходных параметров.

Для алгоритма параллельной коммуникации с барьером (PCB) также оптимальными могут являться только формы BR, LR, SC и SR. LR оптимальна только при коэффициентах $\beta_1/\beta_3 = 0{,}1$, $\beta_2/\beta_3 < 1$ и примерно равных мощностях процессоров P и R, значительно больших S. SR оптимальна при соотношении $\beta_1/\beta_3 \leq 2$ и мощностях процессоров P и R больших, чем у процессора S. В других случаях оптимальными формами выступают BR и SC.

Алгоритм последовательной коммуникации с наложением (SCO) дает наилучшие показатели времени выполнения для форм BR, LR, SC и SR. Формы SR и SC оптимальны при значительно большей мощности процессора P по сравнению с процессорами R и S. Для формы SR это утверждение верно только при $\beta_1/\beta_3 >= 2$. Форма LR может являться оптимальной при $\beta_1/\beta_3 < 1$ и мощности процессора P приблизительно равной процессору R, но значительно превышающей мощность процессора S. Форма BR оптимальна при всех остальных исходных условиях.

Для алгоритма параллельной коммуникации с наложением (PCO) оптимальными также являются формы BR, LR, SC и SR. Форма LR оптимальна только при $\beta_1/\beta_3 < 0{,}6$, $\beta_2/\beta_3 =< 1$ и приблизительно равных вычислительных мощностях процессоров P и R. При значениях коэффициента $\beta_1/\beta_3 \leq 2$ и мощностях процессоров P и R значительно больших, чем у процессора S, оптимальна форма SR. Форма SC оптимальна при мощности процессора P, которая значительно превышает мощность процессоров R и S и значения β_1/β_3, $\beta_2/\beta_3 \geq 1$. Форма BR оптимальна для других случаев.

Следовательно, как и при использовании трехпроцессорных систем с одинаковой пропускной способностью между процессорами [1], традиционные прямоугольные формы разбиения и формы разбиения «прямоугольный угол» (SC) являются оптимальными для любого набора параметров.

Наиболее оптимальной формой разбиения данных между тремя гетерогенными процессорами, объединенными полносвязной топологией, всегда будет являться одна из следующих форм: Square Rectangle, Square Corner, Block Rectangle. Кроме того, Square Corner является оптимальной формой для гетерогенных систем с одним быстрым процессором и двумя медленными; Square Rectangle – для гетерогенных систем с двумя быстрыми процессорами и одним относительно медленным; Block Rectangle – для гетерогенных систем с быстрым, средним и медленным процессором, а также для однородных систем. Форма L-Rectangle может являться оптимальной при мощности процессоров P и R значительно больших, чем у процессора S, и приблизительно равных друг другу при $\beta_1/\beta_3 < 1$ для алгоритма SCB и $\beta_1/\beta_3 = 0{,}1$ для алгоритма PCB. Таким образом, для решения задачи в данном случае достаточно двух процессоров.

Заключение. Web-приложение, разработанное в рамках проведенного исследования, является применимым в широком диапазоне областей знаний, таких как тео-

рия сетей, моделирование физических систем, искусственный интеллект и машинное обучение, транспортные системы.

На основе данных о текущих параметрах вычислительной системы может быть выбран класс алгоритма и оптимальная форма разбиения элементов матрицы между тремя гетерогенными процессорами, объединенными полносвязной топологией, которые позволят добиться наилучших показателей времени выполнения алгоритма.

ЛИТЕРАТУРА

1 DeFlumere, A. // Optimal Partitioning for Parallel Matrix Computation on a Small Number of Abstract Heterogeneous Processors, – 2014. – С. 30-37.

2 Клюева Е.Г., Адамов А.А., Оспанова А.Е., Сницарь Л.Р., Кулбаева Л.Н. Исследование оптимальной формы разбиения данных для умножения матриц на трех гетерогенных процессорах с полносвязной топологией и различными пропускными способностями // Современные наукоемкие технологии. – 2019. – № 2 – С. 83-88.

3 Determination of the optimal shape of matrix elements partitioning on three abstract heterogeneous processors / Y. G. Klyuyeva, V. V. Yavorskij, Adamov [и др.]. – Текст : непосредственный // Cogent Engineering . – 2020. – № 7:1. – С. 1-13.

4 June 2021. – Текст : электронный // Top500 The List : [сайт]. – URL: https://www.top500.org/ (дата обращения: 30.08.2021).

5 DeFlumere A. Optimal Partitioning for Parallel Matrix Computation on a Small Number of Abstract Heterogeneous Processors. PhD thesis, University College Dublin. 2014. – С. 161.

6 Zhong Z., Rychkov V., Lastovetsky A. Data partitioning on heterogeneous multicore platforms. Cluster Computing (CLUSTER), 2011 IEEE International Conference, IEEE. 2011. – С. 580–584.

REFERENCES

1 DeFlumere, A. // Optimal Partitioning for Parallel Matrix Computation on a Small Number of Abstract Heterogeneous Processors, – 2014. – S. 30-37.

2 Klyueva E.G., Adamov A.A., Ospanova A.E., Snicar’ L.R., Kulbaeva L.N. Issledovanie optimal’noj formy razbieniya dannyh dlya umnozheniya matric na trekh geterogennyh processorah s polnosvyaznoj topologiej i razlichnymi propusknymi sposobnostyami // Sovremennye naukoemkie tekhnologii. – 2019. – № 2 – S. 83-88.

3 Determination of the optimal shape of matrix elements partitioning on three abstract heterogeneous processors / Y. G. Klyuyeva, V. V. Yavorskij, Adamov [i dr.]. – Tekst : neposredstvennyj // Cogent Engineering . – 2020. – № 7:1. —– S. 1-13.

4 June 2021. – Tekst : elektronnyj // Top500 The List : [sajt]. – URL: https://www.top500.org/ (data obrashcheniya: 30.08.2021).

5 DeFlumere A. Optimal Partitioning for Parallel Matrix Computation on a Small Number of Abstract Heterogeneous Processors. PhD thesis, University College Dublin. 2014. – S. 161.

6 Zhong Z., Rychkov V., Lastovetsky A. Data partitioning on heterogeneous multicore platforms. Cluster Computing (CLUSTER), 2011 IEEE International Conference, IEEE. 2011. – S. 580–584.

Е. Г. КЛЮЕВА

Караганды техникалық университеті,
Қарағанды, Қазақстан

ҮШ ГЕТЕРОГЕНДІ ПРОЦЕССОРҒА АРТТЫРУ ҮШІН ҮЛКЕН ӨЛШЕМДІ МАТРИЦА ЭЛЕМЕНТТЕРІН БӨЛУДІҢ ОҢТАЙЛЫ ФОРМАСЫН ТАБУҒА АРНАЛҒАН ҚОСЫМШАНЫ ӘЗІРЛЕУ

Мақалада матрица элементтерінің үш дерексіз гетерогенді процессорлары арасында оларды көбейту үшін бөлудің оңтайлы формасын анықтауға веб-қосымшаны әзірлеу нәтижелері келтірілген. Мақалада матрицаны параллель көбейту алгоритмдерінің бес класы берілген: тосқауылмен тізбектей байланыс (Serial Communication with Barrier, SCB), тосқауылмен параллель байланыс (Parallel Communication with Barrier, PCB), қабаттасумен тізбектей байланыс (Serial Communication with Bulk Overlap, SCO), қабаттасумен параллель байланыс (Parallel Communication with Bulk Overlap, PCO), кезектесумен параллельді қабаттасу (Parallel Interleaving Overlap, PIO). Қарастырылып отырған алгоритмдердің коммуникациялық күрделілігін анықтау үшін Хокни моделі қолданылды. Зерттеуде Эшли де Флюмиердің матрица элементтерін процессорлар арасында қайта бөлу үшін «push» технологиясын қолдану барысында анықталған алты тікбұрышты емес элементтерді бөлу формаларын (Square Corner, Rectangle Corner, Square Rectangle, Block Rectangle, L-Rectangle, Traditional 1D Rectangular) қолданады [1]. Белгілі бір техникалық жағдайларда форманың оңтайлылығын анықтау үшін жұмыстарда сипатталған математикалық модельдер қолданылады [2, 3]. Веб-қосымшаны жүзеге асыру үшін Python және JavaScript бағдарламалау тілдері, PIP пакет менеджері және Ajax технологиясы таңдалды. Процессорлар арасындағы матрица элементтерін оңтайлы бөлу мәселесін шешу үлкен өлшемді матрицаларды көбейтуді қолдана отырып, әртүрлі ғылыми салаларда қолданбалы есептерді шешу үшін есептеу ресурстарын тиімді бөлуге мүмкіндік береді.

Түйін сөздер*: Хокни моделі, матрицаларды көбейту, параллель бағдарламалау, деректерді бөлу, гетерогенді параллель жүйелер, веб-қосымша.*

YE. G. KLYUYEVA

Karaganda Technical University,
Karaganda, Kazakhstan

DEVELOPMENT OF THE APPLICATION FOR DETERMINATION OF THE OPTIMAL PARTITIONING FORM OF LARGE-DIMENSIONAL MATRIX'S ELEMENTS FOR MULTIPLICATION ON THREE HETEROGENEOUS PROCESSORS

The article presents the results of the development of a web application for finding the optimal form of splitting matrix elements between three abstract heterogeneous processors when performing the operation of their multiplication. The paper considers five classes of parallel matrix multiplication algorithms: serial communication with a barrier, parallel communication with a barrier, serial communication with overlapping, parallel communication with overlapping, parallel overlapping with alternation. The Hockney model is used to estimate the communication complexity of the algorithms. The work uses six non-rectangular candidate partitioning shapes identified by Ashley DeFlumere in her work [1] as a result of applying the «push» technology of redistribution of matrix elements between the processors: Square

Corner, Rectangle Corner, Square Rectangle, Block Rectangle, L-Rectangle, Traditional 1D Rectangular. Determination of the optimality of the form is made on the basis of mathematical models presented in the works [2,3]. The programming languages Python and JavaScript, the Django framework, the pip package manager, and Ajax technology were used to develop a web application. Solving the problem of determining the optimal matrix shape will allow for efficient planning of computing power resources for various scientific fields using parallel matrix multiplication.

***Keywords**: Hockney model, matrix multiplication, parallel computing, data partitioning, heterogeneous parallel systems, web-application.*