# Раздел II. Алгоритмы обработки информации

УДК 004.273 DOI 10.18522/2311-3103-2022-6-72-83

**И.И. Левин, А.В. Подопригора**

## МЕТОД РАСПАРАЛЛЕЛИВАНИЯ ПО БАЗОВЫМ МАКРООПЕРАЦИЯМ ДЛЯ ОБРАБОТКИ БОЛЬШИХ РАЗРЕЖЕННЫХ НЕСТРУКТУРИРОВАННЫХ МАТРИЦ НА РВС

*Анализ методов и средств обработки больших разреженных неструктурированных матриц кластерными вычислительными системами с традиционной архитектурой показал, что для большинства задач обработки матриц с числом строк порядка $10^5$ производительность снижается в 5-7 раз по сравнению с пиковой производительностью, при этом пиковая производительность вычислительных систем, главным образом, оценивается тестом LINPAC, который предполагает выполнение матричных операций. Основной целью работы является повышение эффективности обработки больших разреженных неструктурированных матриц, для чего целесообразно использовать реконфигурируемые вычислительные системы на основе ПЛИС как основной тип вычислительных средств. Для эффективной обработки больших разреженных неструктурированных матриц на реконфигурируемых вычислительных системах используется ряд ранее описанных в работах методов и подходов, такие как структурная организация вычислений, формат представления больших разреженных неструктурированных матриц «ряд строк», парадигма дискретно-событийной организации потоков данных, метод распараллеливание по итерациям. В статье рассматривается метод распараллеливания по базовым макрооперациям для решения задачи обработки больших разреженных неструктурированных матриц на РВС, который предполагает получение постоянной эффективности вычислений независимо от портрета обрабатываемых больших разреженных неструктурированных матриц. Использование для реконфигурируемых вычислительных систем разработанных методов обработки больших разреженных неструктурированных матриц позволяют обеспечивать эффективность вычислений на уровне 50%, что в несколько раз превосходит эффективность традиционных вычислительных систем.*

*Большие разреженные неструктурированные матрицы; БРН-матрицы; реконфигурируемые вычислительные системы; ПЛИС-технологии; операции над разреженными матрицами; сложение разреженных матриц; умножение разреженных матриц.*

**I.I. Levin, A.V. Podoprigora**

## METHOD OF PARALLELIZATION ON BASIC MACRO OPERATIONS FOR PROCESSING LARGE SPARSE UNSTRUCTURED MATRIXES ON RCS

*Analysis calculating large sparse unstructured matrices (LSU-matrices) methods and tools for cluster computing systems with a traditional architecture showed that for most tasks of processing matrices with about $10^5$ rows, performance compose reduced 5-7 times compared to the peak performance. Meanwhile peak performance of computing systems is mainly estimated by the LINPAC test, which involves the execution of matrix operations. The main goal of the work is to increase the efficiency processing LSU-matrices, for this purpose advisable to use reconfigurable computing systems (RSC) based on FPGAs as the main type of computing tools. For efficient processing LSU-matrices on RCS, a set method and approaches previously described in the papers are used, such as the structural organization of calculations, the format for representing LSU-matrices "row of lines", the paradigm of discrete-event organization of data flows, the meth-*

*od of parallelization by iterations. The article considers the method of parallelization by basic macro-operations for solving the problem of processing LSU-matrices on RCS, which implies obtaining a constant computational efficiency, regardless of the portrait of processed LSU-matrices. Using developed methods for processing LSU-matrices for reconfigurable computing systems makes it possible to provide computational efficiency at the level of 50%, which is several times superior to traditional parallelization methods.*

*Large sparse unstructured matrices; LRN matrices; reconfigurable computing systems; FPGA technologies; sparse matrix operations; sparse matrix addition; sparse matrix multiplication.*

**Введение.** Для повышения эффективности обработки больших разреженных неструктурированных матриц [1] (БРН-матриц) целесообразно использовать РВС на основе ПЛИС [2] как основной тип вычислительных средств, поскольку они позволяют подстраивать вычислительный ресурс системы под решаемую задачу. Широкие возможности архитектуры РВС позволяют организовать структуру, которая будет с большей эффективностью обрабатывать БРН-матрицы [3].

Существующие на данный момент методы обработки матриц на РВС не учитывают неструктурированность и сильную разреженность, что приводит к обработке сильно разреженной матрицы как плотной с эффективностью, определяющейся отношением значимых элементов в матрице к их полному количеству [4]. Поэтому возникает необходимость в разработке специальных методов обработки БРН-матриц для РВС, эффективность которой будет значительно выше, чем при использовании существующих методов обработки матриц на РВС, а также кластерных систем для решения такого рода задач.

Для обработки БРН-матрицы в комплексе используются структурная организация вычислений [5], особый формат хранения БРН-матрицы, метод дискретно-событийной организации потоков данных [6], а также метод распараллеливания по итерациям [7].

Для наиболее эффективной обработки многоместных функций с БРН-матрицами на РВС и возникающей скважностью обработки данных, как следствие использования дискретно-событийной организации потоков данных, предлагается использовать метод распараллеливания по базовым макрооперациям, который предполагает разделение выполняемой функции над БРН-матрицами на отдельные базовые макрооперации с числом операндов, равным двум [7]. В процессе анализа простейших матричных операций было выявлено, что используемые макрооперации над БРН-матрицами могут быть представлены типами Кронекера, Адамара и классического умножения матриц.

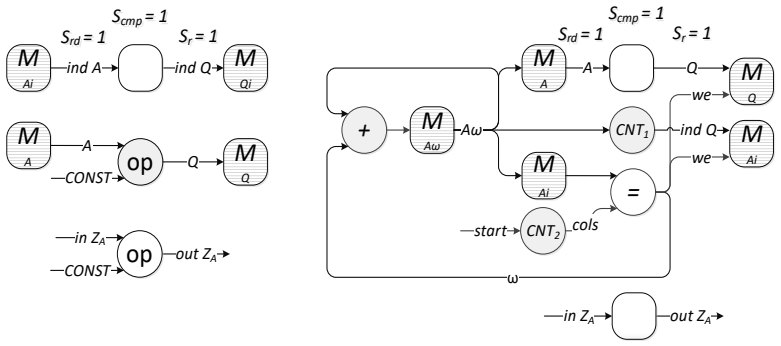
**Операции по типу «Кронекера».** Наиболее простым типом операции являются операции по типу Кронекера, поскольку предполагают изменение одной БРН-матрицы на скалярную величину, которая может быть как отдельной величиной, так и множеством скалярных значений, находящихся в составе вектора или другой матрицы. На практике к таким типам базовых макроопераций могут относиться математические операции типа умножение, сложение, деление, вычитание между БРН-матрицей и скалярной величиной. Для базовых макроопераций типа Кронекера нет необходимости в анализе позиций значимых элементов БРН-матрицы.

Отдельно необходимо выделить операцию транспонирования, поскольку она соответствует типу Кронекера по определяющей характеристике - одна обрабатываемая БРН-матрица. При этом операция транспонирования существенно отличается от арифметических операций с БРН-матрицей и скалярной величиной, тем что изменяет не значимые элементы, а позиции значимых элементов. Для выполнения транспонирования без использования времязатратной итерационной схемы, когда для транспонирования необходимо читать матрицу количество раз, равное количеству столбцов обрабатываемой матрицы, можно использовать косвенную

адресацию в виде дополнительной памяти хранения адресов начала строк БРН-матрицы с использованием дискретно-событийных вычислений. В дополнительной памяти хранятся адреса первых значимых элементов в строках, которые после каждого читаемого элемента инкрементируются.

Для всех выделенных типов макроопераций необходимо обеспечить единую интенсивность на всех этапах выполнения макрооперации: поступления $S_{rd}$, обработки $S_{cmp}$ и выдачи $S_r$ данных [8]. Для арифметических операций над БРН-матрицей типа Кронекера не используется дискретно-событийный поток данных, вследствие чего не возникают разрывы в чтении значимых элементов из памяти, а скважности обработки данных вычислительным блоком базовой макрооперации и выдачи результата равны единицы.

На рис. 1 показаны структурные схемы операций над одной матрицей. Для обоих структур характерно наличие памяти хранения исходной матрицы, состоящей из двух одномерных массивов значимых элементов $M_A$ и позиций значимых элементов $M_{Ai}$.




а б

*Рис. 1. Структурная схема операции над одной матрицей:*
*а – операция типа Кронекера; б – операция транспонирования*

Для макрооперации типа Кронекера, показанной на рис. 1,а, массив позиций значимых элементов $M_{Ai}$ остается неизменным и записывается в $M_{Qi}$, над ненулевыми элементами $M_A$ выполняется определенная арифметическая операция «OP» со скалярной величиной const, после чего результат записывается в $M_Q$. При этом можно учитывать незначащие элементы in $Z_A$, передавая ее и модифицировав в out $Z_A$ операцией «OP» и скалярной величиной const.

Для операции транспонирования, показанной на рис. 1,б, используется дополнительная память $M_\omega$, которая хранит номера ячеек памяти $M_{Ai}$ первых элементов в строках. Счетчик $CNT_2$ для каждой этой ячейки прибавляет значение шага, что позволяет вычитывать значимые элементы из памяти $M_A$, представленной в формате ряд строк, по столбцам с последующим переводом к строке. Счетчик $CNT_1$ используется для подсчета конца строк и формирования транспонированных позиций значащих элементов. При этом сами элементы матрицы $M_A$ и незначащие элементы $Z_A$ никак не изменяются.

**Операции по типу «Адамара».** Второй рассматриваемый тип матричной операции – это операции по типу Адамара. Для этого типа матричных операций характерно взаимодействие двух БРН-матриц между собой. В качестве операндов могут выступать пары матрица-матрица и вектор-вектор БРН-типа. Наиболее часто встречающиеся базовые макрооперации такого типа – операции умножения, сложения, деления и вычитания двух БРН-матриц, однако могут быть и другие.

Для базовых матричных макроопераций типа Адамара используется дискретно-событийная организация потоков данных [9]. Производится анализ текущих позиций значимых элементов, на основе которых возникает команда чтения одного или обоих потоков. При равенстве позиций значимых элементов над соответствующими им значимыми элементами выполняется необходимая арифметическая операция, после чего полученный значимый элемент матрицы записывается в результирующую память с соответствующим позицией. После этого формируется команда на чтение следующего элемента для каждой матрицы. При неравенстве позиций, возникает необходимость определения меньшего из элементов по адресу в строке. Логически это объясняться тем, что меньший по значению позиции значимый элемент находится ближе к началу строки и на смежном ему месте другой матрицы находится незначимый элемент. В этом случае над значимым элементом с меньшей позицией выполняется упрощенная логика арифметической операции с нулем и записывается в результирующую память с соответствующим адресом. После этого формируется команда на чтение одной матрицы с обработанным на этом этапе элементом.

Необходимо учитывать, что в процессе выполнения операций над значимыми элементами результат может оказаться нулевым, следовательно, он не должен быть записан в результирующую память. В этом случае происходит сброс позиции полученного результата с переходом к анализу следующей пары элементов.

На рис. 2 показана структурная схема базовой макрооперации типа Адамара над двумя БРН-матрицами.

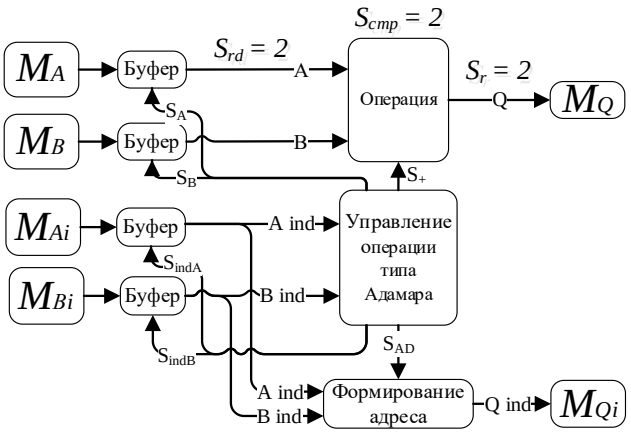



*Рис. 2. Структурная схема базовой макрооперации типа Адамара над двумя БРН-матрицами*

Для этой структуры, как и для рассмотренных ранее, характерно наличие памяти хранения исходных матриц A и B, состоящих из двух одномерных массивов значимых элементов $M_A$ и $M_B$ и позиций значимых элементов матрицы A и матрицы B - $M_{Ai}$ и $M_{Bi}$. Для реализации дискретно-событийных потоков данных значимых элементов матриц и позиций значимых элементов A и B используется «Буфер» между памятями $M_A$, $M_B$, $M_{Ai}$, $M_{Bi}$ и остальной логикой базовой макрооперации. «Буфер» выполняет функцию баланса интенсивности входного потоков относительно интенсивности обработки данных вычислительными блоками операции «Операция» и «формирования адреса».

Буферные элементы позволяют реализовать запуск и остановку чтения элементов из «Буфера» на основании проведенного анализа текущих позиций значимых элементов в блоке «Управление операции типа Адамара» и обеспечить посто-

янную интенсивность потока данных на этапах чтения данных из памяти, их обработку и выдачу результата. Блок «Управление операции типа Адамара» является ключевым, поскольку на основании анализа происходит управление блоком «Операция», отвечающим за выполнение арифметической операции, и блоком «Формирование адреса». После выполнения всех необходимых преобразований с текущими значениями значимых элементов и соответствующих им позиций матриц A и B полученные элемент Q и его позиция Qi записываются в соответствующую результирующую память.

Использование дискретно-событийной модели при организации потоков данных ведет к появлению скважности $S_{cmp}$, соответствующей обработке данных вычислительным блоком базовой макрооперации, пропорциональной количеству участвующих БРН-матриц в операции. В связи с этим блок управления организует разрывы чтения значимых элементов из памяти через буфер $S_{rd}$, что приводит скважность выдачи результирующего элемента $S_r$ к соответствию скважности обработки данных вычислительным блоком $S_{cmp}$.

**Операции типа «умножения матриц».** Последний выделенный тип базовых макроопераций над БРН-матрицами является операцией по типу классического алгоритма умножения матриц. Для этого типа матричных операций характерно взаимодействие двух БРН-матриц между собой. В группе этого типа можно выделить две основные макрооперации - это умножение матрицы на вектор и умножение матрицы на матрицу. Исключением из типа «умножения матриц» является операция умножения вектора на вектор, поскольку результатом является скалярная величина, которая по своему виду не представляет поток данных.

Для базовых матричных макроопераций типа «умножения матриц» используется дискретно-событийная организация потоков данных. Это процесс организации разрывов подачи данных исходного потока используется в точности, как и для базовых матричных макроопераций типа «Адамара». Это происходит за счет управляющих последовательностей блока «Управления операции типа умножения матриц», поступающих в блоки «Буфер». В остальном базовые макрооперации по типу умножения матриц имеют более сложную вычислительную структуру и отличный формат представления БРН-матрицы множителя, которая представлена в разработанном формате список строк, однако в отличие от описанного варианта передается по столбцам.

Типовая структура для операции типа умножения матриц показана на рис. 3. Для этой структуры, как и для рассмотренных ранее, характерно наличие памяти хранения исходных матриц A и B, состоящей из двух одномерных массивов значимых элементов $M_A$ и $M_B$ и позиций значимых элементов $M_{Ai}$ и $M_{Bi}$. Блок «Управление операцией типа умножение матриц» выполняет функцию анализа позиций значимых элементов обрабатываемых исходных матриц, хранящихся в памяти $M_A, M_B$, $M_{Ai}, M_{Bi}$, и определяет необходимость в получении частичных произведений из обрабатываемых значимых элементов матриц в блоке «Умножение» и накоплении его в блоке «Аккумулятор». Помимо этого, в блоке управления происходит анализ конца обрабатываемых текущих строки матрицы A и столбца матрицы B, которые обозначают завершение формирования текущего элемента и его запись в память результирующей матрицы $M_Q$. Кроме управления арифметическими блоками, происходит вычисление позиции значимого элемента в блоке «Формирование адреса» за счет вычисления номера текущей обрабатываемой строки и столбца и записи рассчитанной позиции вычисленного элемента по формату вида представления матриц в память $M_{Qi}$.

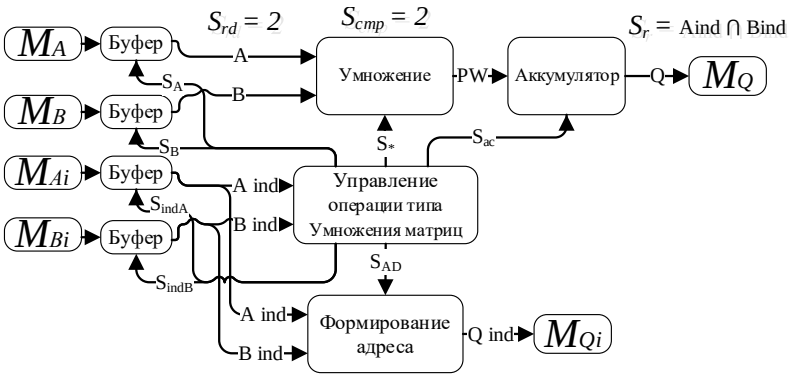



*Рис. 3. Структурная схема базовой макрооперации умножения двух БРН-матриц*

Для базовых макроопераций над БРН-матрицей используется дискретно-событийная модель организации потоков данных, что ведет к появлению скважности, равной двум, на этапе $S_{cmp}$ обработки данных в вычислительном блоке «Умножения» базовой макрооперации. Однако существенной для данного типа базовых макроопераций является скважность выдачи результирующего элемента $S_r$, которая в данном случае определяется количеством пересекающихся позиций значимых элементов строки матрицы A со столбцом матрицы B, что можно обозначить формулой (1), которая описывает формирование скважности для каждого элемента результирующей матрицы за счет пересечения стоки матрицы позиций значимых элементов A со столбцом матрицы позиций значимых элементов:

$$S_r(q_{i,j}) = \sum_{0}^{n} P_k \text{, где } \begin{cases} P_k = 1, \text{если } a_{i,k} \neq 0 \quad \text{и} \quad b_{k,j} \neq 0 ; \\ P_k = 0, \text{если } a_{i,k} = 0 \text{ или } b_{k,j} = 0 . \end{cases} \qquad (1)$$

$S_r(q_{i,j})$ – скважность получения элемента $q_{i,j}$ для операции типа умножения матриц; $P_k$ – показатель совпадения позиций значимых элементов для анализируемых значений; $a_{i,k}$ – элемент матрицы A, анализируемой по срокам; $b_{k,j}$ – элемент матрицы B, анализируемой по столбцам.

В таком случае скважность для получения каждого элемента будет изменяться в зависимости от обрабатываемых строк матрицы A и столбцов матрицы B, что ведет к необходимости использованию методов баланса интенсивностей потоков данных внутри и между базовыми макрооперациями для БРН-матриц.

**Баланс интенсивности потоков данных.** Баланс интенсивности потоков данных обеспечивает синхронизацию интенсивности чтения с интенсивностью записи данных при их неравенстве [10]. Несоответствие интенсивностей в структуре базовой макрооперации возникает на этапе чтения значимых элементов из памяти хранения в исходной памяти и обработки данных вычислительным блоком. Для их балансировки предлагается использование буферных блоков накопления значимых элементов, как показано на структурных схемах базовых макроопераций по типу Адамара на рис. 2 и по типу умножения матриц на рис. 3.

Коэффициент скважности обработки данных по каждой матрице, участвующей в макрооперации, определяется схожим образом для каждой из рассматриваемых типов операций и в общем итоге будет находиться в диапазоне от единицы до двух, включая пограничные значения. Такое нецелочисленное представление скважности обработки БРН-матрицы определяется отношением числа значащих элементов у наибольшей по их значению между матрицей A или B к количеству совпадений позиций значимых элементов матрицы A с матрицей B, что показано в формуле (2).

$$S(q_{cosnt,j}) = \frac{\max(a_{const,j}; b_{const,j})}{\sum_0^n P_k}, \quad (2)$$

где $\begin{cases} P_k = 1, \text{если } a_{const,j} \neq 0 \quad \text{и} \quad b_{const,j} \neq 0 ; \\ P_k = 0, \text{если } a_{const,j} = 0 \text{ или } b_{const,j} = 0 . \end{cases}$

$S(q_{const,j})$ – скважность потока данных в процессе получения результата строки номер const; $P_k$ – показатель совпадения позиций значимых элементов для анализируемой строки; $a_{const,j}$ – значимые элементы матрицы A строки номер const; $b_{const,j}$ – значимые элементы матрицы A строки номер const.

Полученные значения скважности применяются для оценки интенсивности чтения значимых элементов из памяти и интенсивности обработки данных вычислительным блоком. Эти значения используются для расчета минимально необходимой глубины буферных блоков, доступных в вычислительной системе так, чтобы вероятность их переполнения при обработке БРН-матриц была нулевой. Решение этой задачи осуществляется методами сетевых технологий, где возникает вероятность потери пакетов в процессе передачи данных по высокосортным сетям. Применительно к базовым матричным операциям переполнение ведет к прерыванию чтения значимых элементов из памяти и увеличению времени обработки БРН-матриц. Использование формулы стационарной вероятности процесса гибели и размножения [11] при обозначенной размерности, разреженности, среднего количества элементов в строке БРН-матрицы планировать минимально необходимый размер буферных элементов, который обеспечит нулевую вероятность потери значимого элемента.

Помимо использования буферных элементов, для синхронизации интенсивности чтения значимых элементов из памяти хранения исходной матрицы $S_{rd}$ с интенсивностью обработки данных вычислительным блоком $S_{cmp}$ они используются для синхронизации выдачи результирующих элементов. Такая необходимость возникает на стыке последовательно объединенных базовых макроопераций типа умножение матриц. Для сохранения единой интенсивности обработки данных между базовыми макрооперациями типа умножение матриц буфер выдачи результирующих элементов первой операции накапливает значения целой строки, поскольку она является минимальной частью для старта обработки операций типа умножение матриц. Функционально такой подход является частью подхода по объединению базовых матричных операций в вычислительную структуру.

**Создание вычислительной структуры из базовых макроопераций.** Для организации многоместных функций над разреженными матрацами и решения СЛАУ итерационными методами с несколькими БРН-матрицами для разного рода прикладных задач [12] необходимо обозначить подходы по соединению разработанных БРН-матричных макроопераций между собой для реализации структуры многоместной функции или итераций.

Выполнение операции типа «Кронекер» в многоместной функции с другими макрооперациями других типов не требует дополнительных подходов, поскольку вычислительная структура макрооперации позволяет выводить поток данных с той интенсивностью, которая была на входе базовой макрооперации. Необходимость использования специальных подходов соединения возникает для операций типа «Адамар» и «умножение матриц». Для них можно выделить две категории таких подходов – соединение разнотипных и однотипных базовых макроопераций [13].

Основной подход выполнения разнотипных базовых макроопераций над БРН-матрицами основывается на включении операции с меньшей скважностью выдачи результирующего элемента в операцию с большим значением, что предполагает включение базовой макроопераций типа Кронекера или типа Адамара в

базовую макрооперацию типа умножения матриц. Этот подход предполагает преобразование частичных произведений операции типа умножения матриц, операцией типа Адамара, или использования свободного времени в процессе накопления частичных произведений операции типа умножения матриц для выполнения операций по типу Кронекера.

Для создания вычислительной структуры из однотипных базовых макроопераций используется подход, основанный на последовательном или пирамидальном соединении операций. Базовая макрооперация типа Адамара имеет постоянную скважность на всех уровнях макрооперации, а также одинаковую передачу обрабатываемых БРН-матриц в виде формата списка строк. Это позволяет соединять базовые макроопераций такого типа между собой как последовательно, так и пирамидально.

Для соединения двух макроопераций типа умножения матриц существуют особенность в виде передачи одной БРН-матрицы в формате список строк. Вторая БРН-матрица использует такой же формат, но пара векторов формируются по столбцам. Так, с одной загруженной в базовую макрооперацию строкой БРН-матрицы по первому входу и чтением всех столбцов БРН-матрицы по второму входу формирует результат – строку, которая хранится в буфере выдачи результирующего элемента. Полностью сформированная строка является стартовым элементом для запуска вычислений следующей базовой макрооперации типа умножения матриц.

**Оценка эффективности разработанных методов.** Целью разработки методов обработки БРН-матриц на РВС является повышение эффективности вычислений, которые определяются как соотношение значимых вычислений к общему количеству выполненных операций. Предлагаемый набор методов для обработки БРН-матриц позволяет выстраивать все вычислительные узлы в определенную последовательность и производить вычисления только над значимыми данными [14]. За счет однократного и последовательного чтения всех элементов БРН-матрицы из начальной памяти и их проход через вычислительную структуру, в отличие от кластерных ВС с выполнением множественного чтения и записи промежуточных значений обработки БРН-матрицы в память, происходит сокращение времени обработки [15].

Для разработанных специальных методов обработки БРН-матриц эффективность определяется формулой (3), которая основывается на скважности обработки данных вычислительным блоком в соответствии с дискретно-событийной организацией потоков данных и отношением количеством локальных пересечений позиций значимых элементов матриц в базовых макрооперациях к максимальному количеству значимых элементов среди всех матриц:

$$E_{\text{спец РВС}} = \frac{\mathrm{C} + \dfrac{(n^2 \cdot sp_{A_1} \cap n^2 \cdot sp_{A_2}) \cap \ldots \cap (n^2 \cdot sp_{A_{k-1}} \cap n^2 \cdot sp_{A_k})}{\max\left(n^2 \cdot sp_{A_1}; n^2 \cdot sp_{A_2}; \ldots; n^2 \cdot sp_{A_k}\right)}}{sk} \qquad (3)$$

$E_{\text{спец РВС}}$ – эффективность разработанных методов; n – размерность обрабатываемых матриц; $sp_{Ak}$ – степень разреженности обрабатываемых матриц; sk – скважность обработки данных, вызванная методом организации дискретно-событийных потоков данных, равная количеству участвующих в базовой макрооперации операндов; *C* – количество одновременно выдаваемых данных.

На рис. 5 показаны графики эффективностей вычислительных системы на задаче обработки БРН-матриц.

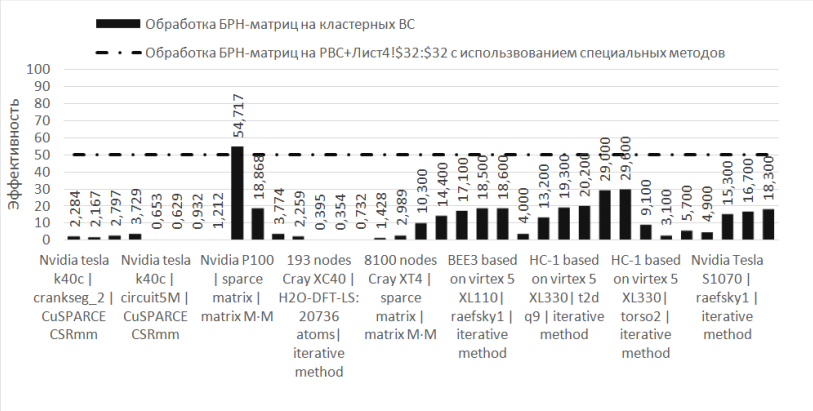



*Рис. 5. График сравнения эффективностей использования кластерных МПС с классической архитектурой и специальных методов обработки БРН-матриц на РВС*

График эффективности классических вычислительных систем, с использованием высокопроизводительных графических ускорителей [16], был получен в ходе анализа работ, посвященных обработке БРН-матриц как отношений практической производительности вычислительной системы к пиковой производительности вычислительной системы, обозначенной ее разработчиком [17–21]. Ось абсцисс графика обозначает рассмотренный набор задач, из которого получены значения эффективности на соответствующих системах, для которых подпись к значениям эффективности определяет основной вычислительный компонент или название вычислительной системы, обрабатываемую матрицу и тип выполняемой операции, функции или решения систем линейных алгебраических уравнений.

График эффективности специальных методов обработки БРН-матриц на РВС, построенный на основании выведенной формулы (3), где sk – скважность обработки данных с использованием распараллеливания по базовым макрооперациям, равная 2, количество одновременно выдаваемых данных $C$ равно 1. Второе слагаемое числителя, представленное дробью, принимает значение 0, как случай, для которого совпадения позиций значимых элементов обрабатываемых БРН-матриц отсутствуют. В результате эффективность на всех задачах обработки БРН-матриц будет на уровне 50% значения.

**Заключение.** Использование для РВС разработанных методов обработки БРН-матриц, включающие в себя ранее описанные методы: формат хранения БРН-матриц «список строк», метода организации дискретно-событийных потоков данных для базовых макроопераций БРН-матрицами, а также описанные в статье процесс формирования базовых макроопераций и их объединения, баланс скважности на этапах базовых макроопераций позволяют обеспечивать наименьшую эффективности вычислений на уровне 50%, что в несколько раз выше традиционные методы распараллеливания. При этом не исключается возможность использования дополнительных ресурсов РВС, на реализацию параллельной обработки нескольких элементов БРН-матриц, за счет чего можно производить опережающий анализ и нивелировать скважность потока обработки данных. В результате эффективность РВС будет находиться в диапазоне 60–85%, относительно пиковой производительности системы.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК


1. *Kolodziej S.P., Aznaveh M., Bullock M., David J., Davis T.A., Henderson M., Hu Y., Sandstrom R.* The SuiteSparse Matrix Collection Website Interface // Journal of Open Source Software. – March 2019. – Vol. 4, No. 35. – P. 1244-1248. – DOI: https://doi.org/10.21105/joss.01244 (дата обращения: 02.10.2022).
2. *Гузик В.Ф., Каляев И.А., Левин И.И.* Реконфигурируемые вычислительные системы / под ред. И.А. Каляева. – Таганрог: Изд-во ЮФУ, 2016. – 472 с.
3. *Дордопуло А.И., Каляев И.А., Левин И.И., Семерников Е.А.* Семейство многопроцессорных вычислительных систем с динамически перестраиваемой архитектурой // Многопроцессорные вычислительные и управляющие системы: Матер. научно-технической конференции. – Таганрог, 2007. – С. 11-17.
4. *Пелипец А.В.* Методы и средства решения задач линейной алгебры на высокопроизводительных реконфигурируемых вычислительных системах: дисс. … канд. техн. наук. – Таганрог, 2016. – 199 с.
5. *Каляев И.А., Левин И.И., Семерников Е.А., Шмойлов В.И.* Реконфигурируемые мультиконвейерные вычислительные структуры / под общ. ред. И.А. Каляева. – 2-е изд. перераб. и доп. – Ростов-на-Дону: Изд-во ЮНЦ РАН, 2009. – 344 с.
6. *Подопригора А.В.* Метод организации дискретно-событийных вычислений для обработки больших разреженных неструктурированных матриц на РВС // Известия ЮФУ. Технические науки. – 2021. – № 7. – С. 189-197. – DOI 10.18522/2311-3103-2021-7-189-197.
7. *Подопригора А.В.* Методы распараллеливания вычислений для обработки больших разреженных неструктурированных матриц на РВС // XVIII Ежегодная молодежная научная конференция «Наука Юга России: достижения и перспективы»: Матер. конференции (г. Ростов-на-Дону, 18–29 апреля 2022 г.). – Ростов-на-Дону: Изд-во ЮНЦ РАН, 2022. – С. 262. – ISBN 978-5-4358-0233-7.
8. *Сорокин Д.А*. Методы решения задач с переменной интенсивностью потоков данных на реконфигурируемых вычислительных системах: дисс. … канд. техн. наук: 05.13.11: защищена 15.06.12: утверждена: 11.03.13. – Таганрог, 2013. – 168 с. – 005043774.
9. *Подопригора А.В.* Управление процессом обработки разреженных матриц в дискретно-событийных матричных операциях // XIV Всероссийская мультиконференция по проблемам управления (МКПУ-2021): Матер. XIV мультиконференции (Дивноморское, Геленджик, 27 сентября – 2 октября 2021 г.): в 4 т. Т. 2 / редкол.: И.А. Каляев, В.Г. Пешехонов и др. – Ростов-на-Дону; Таганрог: Изд-во ЮФУ, 2021.– С. 276-278. – ISBN 978-5-9275-3846-1.
10. *Клейнрок Л.* Теория массового обслуживания. – М.: Машиностроение, 1979. – 432 с.
11. *Коннов А.Л., Ушаков Ю.А.* Методы расчета показателей производительности сетей ЭВМ с неоднородным трафиком. – Оренбург: ОГУ, 2013. – С. 10-16.
12. *Тихонов А.Н., Самарский А.А.* Уравнения математической физики. – М.: Изд-во Московского университета, 1999. – 6-е изд. – 798 с. – URL: https://elar.urfu.ru/bitstream/10995/42951/1/978-5-321-02475-1_2016.pdf (дата обращения: 15.10.2022).
13. *Подопригора А.В.* Объединение базовых БРН-матричных макроопераций // Многопроцессорные вычислительные и управляющие системы: Матер. Всероссийской научно-технической конференции (г. Таганрог 27–30 июня 2022 г). – Ростов-на-Дону – Таганрог: Изд-во ЮФУ, 2022. – С. 103. – ISBN 978-5-9275-4144-7.
14. *Подопригора А.В., Чекина М.Д.* Решение разреженных СЛАУ большой и сверхбольшой размерности многосеточным методом на РВС // Известия ЮФУ. Технические науки. – 2018. – № 8. – С. 212-218. – DOI: 10.23683/2311-3103-2018-8-212-221.
15. *Каляев А.В., Левин И.И.* Модульно-наращиваемые многопроцессорные системы со структурно-процедурной организацией вычислений. – М.: Янус-К, 2003. – 380 с.
16. Параллельные вычисления CUDA / NVIDIA Corporation. – 2018. – URL: http://www.nvidia.ru/object/cuda-parallel-computing-ru.html (дата обращения: 18.10.2022).
17. *Bethune I., Gloss A., Hutter J., Lazzaro A., Pabst H., Reid F.* Porting of the DBCSR library for Sparse Matrix-Matrix Multiplications to Intel Xeon Phi systems // Submitted to the ParCo2017 conference. Distributed, Parallel, and Cluster Computing (cs.DC) - 2017 Italy Bologna 12-15 September 2017. – DOI: 10.3233/978-1-61499-843-3-47.

18. *Chungz E.S., Davisz J.D., Kestury S.* An FPGA Drop-In Replacement for Universal Matrix-Vector Multiplication. – Portland: Workshop on the Intersections of Computer Architecture and Reconfigurable Logic, 2012. – P. 1-6.
19. *Georgopoulos L., Sobczyk A., Christofidellis D., Dolfi M., Auer C., Staar P., Bekas C.* Enhancing multi-threaded sparse matrix multiplication for knowledge graph-oriented algorithms and analytics IBM Research. – Zurich Säumerstrasse 4 CH-8803 Rüschlikon Switzerland 2019. – 11 p.
20. *Kunchum R.* On Improving Sparse Matrix-Matrix Multiplication on GPUs (Thesis). The Ohio State University, 2017. – P. 36-42. – https://etd.ohiolink.edu/!etd.send_file?accession=osu1492694387445938&disposition=inline.
21. *Yang C., Buluc A., Owens J.* Design Principles for Sparse Matrix Multiplication on the GPU // International European Conference on Parallel and Distributed Computing. Turin, 2018. – P. 12.

REFERENCES

1. *Kolodziej S.P., Aznaveh M., Bullock M., David J., Davis T.A., Henderson M., Hu Y., Sandstrom R.* The SuiteSparse Matrix Collection Website Interface, *Journal of Open Source Software,* March 2019, Vol. 4, No. 35, pp. 1244-1248. DOI: https://doi.org/10.21105/joss.01244 (accessed 02 October 2022).
2. *Guzik V.F., Kalyaev I.A., Levin I.I.* Rekonfiguriruemye vychislitel'nye sistemy [Reconfigurable computing systems], ed. by I.A. Kalyaeva. Taganrog: Izd-vo YuFU, 2016, 472 p.
3. *Dordopulo A.I., Kalyaev I.A., Levin I.I., Semernikov E.A.* Semeystvo mnogoprotsessornykh vychislitel'nykh sistem s dinamicheski perestraivaemoy arkhitekturoy [A family of multiprocessor computing systems with dynamically tunable architecture], *Mnogoprotsessornye vychislitel'nye i upravlyayushchie sistemy: Mater. nauchno-tekhnicheskoy konferentsii* [Multiprocessor computing and control systems: Proceedings of the scientific and technical conference]. Taganrog, 2007, pp. 11-17.
4. *Pelipets A.V.* Metody i sredstva resheniya zadach lineynoy algebry na vysokoproizvoditel'nykh rekonfiguriruemykh vychislitel'nykh sistemakh: disc. … kand. tekhn. nauk [Methods and means of solving linear algebra problems on high-performance reconfigurable computing systems: cand. of eng. sc. diss.]. Taganrog, 2016, 199 p.
5. *Kalyaev I.A., Levin I.I., Semernikov E.A., Shmoylov V.I.* Rekonfiguriruemye mul'tikonveyernye vychislitel'nye struktury [Reconfigurable multiconveyor computing structures], under the general editorship of I.A. Kalyaev. 2nd ed. Rostov-on-Don: Izd-vo YuNTS RAN, 2009, 344 p.
6. *Podoprigora A.V.* Metod organizatsii diskretno-sobytiynykh vychisleniy dlya obrabotki bol'shikh razrezhennykh nestrukturirovannykh matrits na RVS [A method for organizing discrete-event computing for processing large sparse unstructured matrices on RVS], I*zvestiya YuFU. Tekhnicheskie nauki* [Izvestiya SFedU. Engineering Sciences], 2021, No. 7, pp. 189-197. DOI: 10.18522/2311-3103-2021-7-189-197.
7. *Podoprigora A.V.* Metody rasparallelivaniya vychisleniy dlya obrabotki bol'shikh razrezhennykh nestrukturirovannykh matrits na RVS [Methods of parallelization of calculations for processing large sparse unstructured matrices on RVS], *XVIII Ezhegodnaya molodezhnaya nauchnaya konferentsiya «Nauka YUga Rossii: dostizheniya i perspektivy»: Mater. konferentsii (g. Rostov-na-Donu, 18–29 aprelya 2022 g.)* [XVIII Annual Youth Scientific Conference "Science of the South of Russia: achievements and prospects": Materials of the conference (Rostov-on-Don, Rostov–on-Don, April 18-29, 2022)]. Rostov-on-Don: Izd-vo YuNTS RAN, 2022, pp. 262. ISBN 978-5-4358-0233-7.
8. *Sorokin D.A.* Metody resheniya zadach s peremennoy intensivnost'yu potokov dannykh na rekonfiguriruemykh vychislitel'nykh sistemakh: diss. … kand. tekhn. nauk [Methods for solving problems with variable intensity of data flows on reconfigurable computing systems: cand. of eng. sc. diss.]: 05.13.11: protected 15.06.12: approved: 11.03.13. Taganrog, 2013, 168 p. 005043774.
9. *Podoprigora A.V.* Upravlenie protsessom obrabotki razrezhennykh matrits v diskretno-sobytiynykh matrichnykh operatsiyakh [Managing the process of processing sparse matrices in discrete-event matrix operations], *XIV Vserossiyskaya mul'tikonferentsiya po problemam upravleniya (MKPU-2021): Mater. XIV mul'tikonferentsii (Divnomorskoe, Gelendzhik, 27 sentyabrya – 2 oktyabrya 2021 g.)* [XIV All-Russian Multi-conference on Management Problems (MCPU-2021): Proceedings of the XIV multi-conference (Divnomorskoe, Gelendzhik, September 27 – October 2, 2021)]: in 4 vol. Vol. 2, editorial board: I.A. Kalyaev, V.G. Peshekhonov and others. Rostov-on-Don; Taganrog: Izd-vo YuFU, 2021., pp. 276-278. ISBN 978-5-9275-3846-1.

10. *Kleynrok L.* Teoriya massovogo obsluzhivaniya [Theory of queuing]. Moscow: Mashinostroenie, 1979, 432 p.
11. *Konnov A.L., Ushakov Yu.A.* Metody rascheta pokazateley proizvoditel'nosti setey EVM s neodnorodnym trafikom [Methods for calculating performance indicators of computer networks with heterogeneous traffic]. Orenburg: OGU, 2013, pp. 10-16.
12. *Tikhonov A.N., Samarskiy A.A.* Uravneniya matematicheskoy fiziki [Equations of mathematical physics]. Moscow: Izd-vo Moskovskogo universiteta, 1999. 6th ed., 798 p. Available at: https://elar.urfu.ru/bitstream/10995/42951/1/978-5-321-02475-1_2016.pdf (accessed 15 October 2022).
13. *Podoprigora A.V.* Ob"edinenie bazovykh BRN-matrichnykh makrooperatsiy [Combining basic BRN-matrix macro operations], *Mnogoprotsessornye vychislitel'nye i upravlyayushchie sistemy: Mater. Vserossiyskoy nauchno-tekhnicheskoy konferentsii (g. Taganrog 27–30 iyunya 2022 g)* [.Multiprocessor computing and control systems: Materials of the All-Russian Scientific and Technical Conference (Taganrog, June 27-30, 2022)]. Rostov-on-Don – Taganrog: Izd-vo YuFU, 2022, pp. 103. ISBN 978-5-9275-4144-7.
14. *Podoprigora A.V., Chekina M.D.* Reshenie razrezhennykh SLAU bol'shoy i sverkhbol'shoy razmernosti mnogosetochnym metodom na RVS [The solution of sparse SLOWS of large and extra-large dimensions by the multigrid method on RVS], *Izvestiya YuFU. Tekhnicheskie nauki* [Izvestiya SFedU. Engineering Sciences], 2018, No. 8, pp. 212-218. DOI: 10.23683/2311-3103-2018-8-212-221.
15. *Kalyaev A.V., Levin I.I.* Modul'no-narashchivaemye mnogoprotsessornye sistemy so strukturno-protsedurnoy organizatsiey vychisleniy [Modular-stackable multiprocessor systems with structural and procedural organization of computing]. Moscow: Yanus-K, 2003, 380 p.
16. Parallel'nye vychisleniya CUDA, NVIDIA Corporation [Parallel computing CUDA, NVIDIA Corporation], 2018. Available at: http://www.nvidia.ru/object/cuda-parallel-computing-ru.html (accessed 18 October 2022).
17. *Bethune I., Gloss A., Hutter J., Lazzaro A., Pabst H., Reid F.* Porting of the DBCSR library for Sparse Matrix-Matrix Multiplications to Intel Xeon Phi systems, *Submitted to the ParCo2017 conference. Distributed, Parallel, and Cluster Computing (cs.DC) - 2017 Italy Bologna 12-15 September 2017*. DOI: 10.3233/978-1-61499-843-3-47.
18. *Chungz E.S., Davisz J.D., Kestury S.* An FPGA Drop-In Replacement for Universal Matrix-Vector Multiplication. Portland: Workshop on the Intersections of Computer Architecture and Reconfigurable Logic, 2012, pp. 1-6.
19. *Georgopoulos L., Sobczyk A., Christofidellis D., Dolfi M., Auer C., Staar P., Bekas C.* Enhancing multi-threaded sparse matrix multiplication for knowledge graph-oriented algorithms and analytics IBM Research. Zurich Säumerstrasse 4 CH-8803 Rüschlikon Switzerland 2019, 11 p.
20. *Kunchum R.* On Improving Sparse Matrix-Matrix Multiplication on GPUs (Thesis). The Ohio State University, 2017, pp. 36-42. Available at: https://etd.ohiolink.edu/!etd.send_file? accession=osu1492694387445938&disposition=inline.
21. *Yang C., Buluc A., Owens J.* Design Principles for Sparse Matrix Multiplication on the GPU, *International European Conference on Parallel and Distributed Computing. Turin, 2018*, pp. 12.

Статью рекомендовал к опубликованию д.т.н. Э.В. Мельник.

**Левин Илья Израилевич** – Южный федеральный университет; e-mail: iilevin@sfedu.ru; г. Таганрог, Россия; зав. кафедрой ИМС; д.т.н.; профессор.

**Подопригора Александр Владимирович** – e-mail: apodoprigora@sfedu.ru; аспирант.

**Levin Il'ya Izrailevich** – Southern Federal University; e-mail: iilevin@sfedu.ru; Taganrog, Russia; head the department; dr. of eng. sc.; professor.

**Podoprigora Aleksander Vladimirovich** – e-mail: apodoprigora@sfedu.ru; postgraduate student.