

УДК 621.315; 004.051

## Оценка загрузки гибридного вычислительного комплекса при выполнении задач моделирования в материаловедении

© 2020 г. К. И. Волович<sup>1,§</sup>

<sup>1</sup> *Федеральный исследовательский центр «Информатика и управление»  
Российской академии наук,  
ул. Вавилова, д. 44, корп. 2, Москва, 119333, Россия*

**Аннотация.** Статья посвящена методикам расчета и оценке эффективности функционирования гибридных вычислительных систем. Программные системы материаловедения демонстрируют максимальную эффективность при функционировании именно на гибридных вычислительных системах при использовании графических ускорителей для проведения расчетов. В качестве примера можно привести программные системы VASP (The Vienna Ab initio Simulation Package) и Quantum ESPRESSO. Эти программные системы проявляют наибольшую эффективность при монопольном использовании вычислительных ресурсов: RAM, CPU, GPU. При эксплуатации гибридного высокопроизводительного комплекса возникает задача управления ресурсами и разделения их между группой пользователей. Необходимо разработать технологии, которые обеспечивают выделение ресурсов приложениям материаловедения для разных пользователей и научных коллективов. Современным подходом в организации вычислительного процесса является использование технологий виртуализации и облачных технологий. Облачные технологии позволяют предоставлять пользователям услуги SaaS и PaaS. Целесообразно предоставлять научным командам прикладные системы материаловедения как облачные сервисы. Такие разноплановые подходы в условиях применения в одном вычислительном комплексе требуют выработки методов по оптимизации загрузки ресурсов высокопроизводительного комплекса, оценке эффективности использования его вычислительных возможностей и выработке методики совершенствования пользовательских программ.

Определение качества загрузки комплекса является важной задачей при предоставлении сервисов высокопроизводительных вычислений научным коллективам, выполняющими междисциплинарные научные исследования в различных областях науки и техники. В статье предлагается метод расчета значения величины загрузки с использованием пиковых значений производительности комплекса. Анализируются результаты и качество функционирования облачных научных сервисов высокопроизводительных вычислений с помощью Roofline-модели.

**Ключевые слова:** гибридная архитектура, эффективность, оптимизация кода, облачный сервис, графический ускоритель, оценка загрузки, оценка эффективности

### Введение

Важнейшей задачей при эксплуатации высокопроизводительного вычислительного комплекса является обеспечение наиболее полной загрузки его ресурсов.

Задачи материаловедения представляют собой смесь вычислительных заданий, требующих для

эффективного выполнения различных ресурсов вычислительного комплекса: оперативной памяти, ядер центральных процессоров, графических ускорителей вычислений [1, 2].

Необходимо отметить, что в условиях функционирования крупного вычислительного комплекса необходимо обеспечить одновременное решение ряда научных задач, требующих аналогичных вычислительных ресурсов, но предъявляющих другие требования к программному окружению прикладной задачи [3—5].

Для решения междисциплинарных научных задач в том числе в области материаловедения, биомедицинской химии, оптимизации [6], предлагается метод предоставления ресурсов вычислительных комплексов как облачных сервисов цифровой платформы [7—9]. Особенностью функционирования цифровой платформы в части предоставления услуг

**Волович Константин Иосифович**<sup>1,§</sup> — канд. техн. наук, старший научный сотрудник, e-mail: kvolovich@frccsc.ru

<sup>§</sup> Автор для переписки

Статья подготовлена по материалам доклада, представленного на II-й международной конференции «Математическое моделирование в материаловедении электронных компонентов», Москва, 19—21 октября 2020 г. (Волович К.И. Оценка загрузки гибридного вычислительного комплекса при выполнении задач моделирования в материаловедении. М.: МАКС Пресс, 2020. С. 30—33. DOI: 10.29003/m1511.MMMSEC-2020/30-33)

высокопроизводительных вычислений являются повышенные требования к оптимизации функционирования гибридных вычислителей, направленные на увеличение их загрузки пользовательскими задачами [10].

В задаче максимизации загрузки гибридного высокопроизводительного комплекса можно выделить два основных направления [11, 12]:

- обеспечение выполнения максимально возможного количества приложений на определенном периоде времени;

- максимально качественное использование ресурсов комплекса приложениями пользователей.

Определение качества загрузки комплекса является важной задачей при эксплуатации, поскольку позволяет планировать предоставление ресурсов, оценивать необходимость модернизации, определять качество предоставляемых услуг.

Как правило, загрузка определяется как отношение метрики (параметра) загрузки к максимально возможному значению данного параметра. Метрика определяется путем измерения или расчета.

Для повышения эффективности прикладного программного кода следует осуществлять выработку рекомендаций по оптимизации пользовательских программ. Отметим, что особенностью вычислительных заданий материаловедения является активное использование интегрированных программных сред VASP, Quantum ESPRESSO и др. Рекомендации по оптимизации могут касаться как пользовательского программного кода, так и кода интегрированных сред. Использование Roofline-моделей позволяет проанализировать основные программные процедуры прикладного программного обеспечения и интегрированных сред и выработать направления оптимизации программного кода.

### Оценка загрузки гибридного вычислительного комплекса при предоставлении ресурсов в рамках цифровой платформы

Для традиционных суперкомпьютеров параметром загрузки может быть количество ядро-часов, которые были предоставлены приложению для выполнения расчетов [13]. Отношение выделенных ядро-часов к максимально возможному является показателем загрузки комплекса. Данные параметры рассчитываются для определенного периода времени и представляют собой интегральный показатель загрузки за данный период.

Для гибридных архитектур, предоставляющих вычислительные ресурсы как сервисы цифровой платформы, такой подход представляется менее показательным, поскольку в гибридном вычислительном комплексе присутствуют ядра различных типов, а задания резервируют ресурсы вычислителя не ядрами, а целиком графическими ускорителями.

При этом заказ вычислительных сервисов осуществляется разными научными коллективами, размещающих в вычислительном комплексе приложения из различных областей науки и техники, которые требуют различной программной среды и различного состава вычислительных ресурсов.

Задачи выполнения расчетов материаловедения и смежных научных дисциплин, использующих интегрированные системы расчета межмолекулярных взаимодействий, являются ярким примером

Предлагаемая методика позволяет учесть эту особенность. Оценка загрузки определяется путем сравнения запрашиваемого приложениями и максимально возможного количество операций с плавающей точкой за единицу времени.

Отметим, что существует различие между теоретически возможной производительностью компонентов и комплекса в целом (пиковая производительность) и практически достижимыми результатами. Результаты определяются различными тестами и сильно отличающимися в зависимости от типа задач и параметров настройки комплекса [14].

Для оценки загрузки гибридного высокопроизводительного вычислительного комплекса воспользуемся пиковой производительностью. Она определяется как сумма пиковых производительностей его компонентов — узлов

$$P_{\text{peak}} = \sum_{i=1}^K P_{\text{host } i}, \quad (1)$$

где  $P_{\text{peak}}$  — пиковая производительность вычислительного комплекса;  $P_{\text{host } i}$  — пиковая производительность ( $P_{\text{host}}$ )  $i$ -го узла вычислительного комплекса.

Отметим, что при суммировании не учитываются потери производительности, которые возникают при взаимодействии узлов по объединяющей их вычислительной сети (*Interconnect*).

Пиковая производительность узла  $P_{\text{host}}$  определяется как сумма производительностей центральных процессоров узла  $P_{\text{cpu}}$  и его графических ускорителей  $P_{\text{gpu}}$ . Предполагается, что они полностью загружены операциями с плавающей точкой, не выполняют каких-либо других операций, а потери на передачу данных между центральными процессорами и графическими ускорителями отсутствуют

$$P_{\text{host}} = N_{\text{cpu}} P_{\text{cpu}} + N_{\text{gpu}} P_{\text{gpu}}, \quad (2)$$

где  $N_{\text{cpu}}$  — количество центральных процессоров в вычислительном узле;  $N_{\text{gpu}}$  — количество графических ускорителей в вычислительном узле;  $P_{\text{cpu}}$  — пиковая производительность центрального процессора;  $P_{\text{gpu}}$  — пиковая производительность графического ускорителя.

Для расчета пиковой производительности центрального процессора (3) будем считать, что операции выполняются ядрами параллельно, каждое ядро может обрабатывать группу потоков, а в потоке допускается параллельное выполнение нескольких операций, если для этого имеется несколько операционных блоков. Такая ядерно-потоковая архитектура характерна для современных классических процессоров различных производителей:

$$P_{\text{cpu}} = n_{\text{core}} n_{\text{stream}} n_{\text{unit}} F_{\text{cpu}}, \quad (3)$$

где  $n_{\text{core}}$  — количество ядер центрального процессора;  $n_{\text{stream}}$  — количество потоков, обрабатываемых ядром центрального процессора;  $n_{\text{unit}}$  — количество операционных блоков на проток, соответствует количеству операций, выполняемых в одном потоке за один такт;  $F_{\text{cpu}}$  — частота центрального процессора.

Для оценки производительности графических ускорителей воспользуемся свойствами современной архитектурой ускорителей компании NVidia семейства Volta, как наиболее популярных. Ускорители NVidia содержат ядра cuda- и tensor-ядра, позволяющие выполнять параллельные операции над числами с плавающей точкой и матрицами таких чисел. Производительность графического ускорителя определяется как сумма производительностей всех ядер без учета потерь на диспетчеризацию и взаимодействие [15—17]

$$P_{\text{gpu}} = P_{\text{cuda}} + P_{\text{tensor}}, \quad (4)$$

где  $P_{\text{cuda}}$  — суммарная производительность cuda-ядер графического ускорителя;  $P_{\text{tensor}}$  — суммарная производительность tensor-ядер графического ускорителя.

Определим значение производительности cuda-ядер графического ускорителя по формуле (5), считая, что операция с плавающей точкой выполняется за один такт.

Tensor-ядра выполняют за один такт перемножение квадратных матриц. При расчете количества операций, выполняемых при этом, будем учитывать, что вычисление каждого элемента результирующей матрицы требует выполнения операций умножения, равного порядку матрицы, а также операций сложения на единицу меньше. Таким образом, общая производительность tensor-ядер вычисляется по формуле (6).

Отметим, что точность выполнения операций с плавающей точкой для различных ядер может отличаться. В графическом ускорителе NVidia Tesla V 100 cuda-ядра используют числам двойной точности, а tensor — одинарной точности. В данной методике оценки производительности эта особенность не учитывается.

Суммарные производительности cuda- и tensor-ядер определяются по формулам (5) и (6):

$$P_{\text{cuda}} = n_{\text{cuda}} F_{\text{gpu}}, \quad (5)$$

$$P_{\text{tensor}} = n_{\text{tensor}} r^2 (2r - 1) F_{\text{gpu}}, \quad (6)$$

где  $n_{\text{cuda}}$  — количество cuda-ядер графического ускорителя;  $n_{\text{tensor}}$  — количество tensor-ядер графического ускорителя;  $r$  — порядок квадратной матрицы;  $F_{\text{gpu}}$  — частота графического ускорителя.

Пиковая производительность графического ускорителя вычисляется по формуле

$$P_{\text{gpu}} = [n_{\text{cuda}} + n_{\text{tensor}} r^2 (2r - 1)] F_{\text{gpu}}. \quad (7)$$

Пиковая производительность вычислительного узла гибридного высокопроизводительного вычислительного комплекса вычисляется по формуле (8).

$$P_{\text{host}} = N_{\text{cpu}} n_{\text{core}} n_{\text{stream}} n_{\text{unit}} F_{\text{cpu}} + [n_{\text{cuda}} + n_{\text{tensor}} r^2 (2r - 1)] F_{\text{gpu}}. \quad (8)$$

Суммарная пиковая производительность гибридного высокопроизводительного комплекса вычисляется по формуле (1).

Как показано выше, производительность вычислительного комплекса рассчитывается как сумма производительностей его компонентов и выражается количеством операций с плавающей точкой, выполняемых в секунду.

Ресурсом гибридного высокопроизводительного вычислительного комплекса на интервале времени будет пиковое значение количества операций с плавающей точкой, доступное пользователям в течение этого интервала.

Рассмотрим суммарное количество операций гибридного высокопроизводительного вычислительного комплекса  $Op(T)$  на интервале времени  $T$  определяется как:

$$Op(T) = P_{\text{peak}} T, \quad (9)$$

где  $T$  — интервал времени.

Такая пиковая оценка отличается от фактической, получаемой на основе различных тестов. Однако, как отмечалось выше, в данной методике будем пользоваться пиковыми значениями.

Для оценки требований приложений к ресурсам гибридного высокопроизводительного вычислительного комплекса вычислим требуемое для выполнения приложения количество операций (10).

Для каждого приложения резервируется некоторое количество ядер центральных процессоров, графических ускорителей, а также время выполнения. Учтем, что ресурсы графических ускорителей

резервируются целиком, а ресурсы центральных процессоров — по ядрам. Поэтому суммарное количество операций гибридного высокопроизводительного вычислительного комплекса, выполненных задачей  $Op_{app}(t)$  за заданное время  $t$  определяется количеством зарезервированных приложением ядер центральных процессоров ( $R_{cpu}$ ) и графических ускорителей ( $R_{gpu}$ )

$$Op_{app}(t) = \left( \frac{R_{cpu} P_{cpu}}{n} + R_{gpu} P_{gpu} \right) t, \quad (10)$$

где  $R_{cpu}$  — количество зарезервированных приложением ядер центральных процессоров;  $n$  — общее количество ядер центральных процессоров;  $R_{gpu}$  — количество зарезервированных приложением графических ускорителей.

После расчетов для всех приложений  $i = 1, \dots, N$ , выполнение которых приходилось на период  $T$ , получим суммарное количество операций, требуемых для выполнения приложений на периоде  $T$ :

$$Op_{app}(T) = \sum_{i=1}^N \left( \frac{R_{cpu} i P_{cpu}}{n} + R_{gpu} i P_{gpu} \right) t_i, \quad \text{для } t_i \in T. \quad (11)$$

### Исследование производительности вычислительных узлов при предоставлении облачных сервисов

Использование расчетной пиковой производительности вычислительных узлов позволяет оценить требуемый приложениями объем вычислительных операций, однако не позволяет судить о том, насколько эффективно используются ресурсы вычислительного комплекса.

Общей задачей организации вычислительного процесса при предоставлении облачных услуг цифровой платформы научных сервисов является

максимизация реальной производительности вычислительного узла. В суперкомпьютерной области принято оценивать реальную производительность вычислительных комплексов с помощью тестов задач линейной алгебры LINPACK [14]. Однако, показанная на данных тестах производительность как правило не достигается при решении конкретных научных задач. Это связано с недостаточной оптимизацией программного кода прикладных задач и ее адаптации к архитектуре вычислительного комплекса.

Поэтому при эксплуатации вычислительного комплекса полезно производить оценку качества функционирования пользовательских приложений и выдавать рекомендации пользователям по оптимизации приложений.

Качество использования ресурсов вычислительного комплекса приложением полезно представлять с использованием так называемой Roofline-модели [18].

Модель позволяет показать производительность компонентов алгоритма (процедур и циклов) в условиях естественных ограничений вычислительной системы — производительности памяти и пиковой производительности вычислительного блока. Причем, в зависимости от использованной технологии могут устанавливаться различные границы. Так, производительность памяти для собственно RAM и кешей разного уровня является различной. Поэтому при условии использования приложением разных технологий работы с памятью технологические ограничения производительности различаются. Аналогично, производительность вычислительных блоков ограничивается возможностями технологии: скалярные вычисления, векторизация и т. д.

Проведенные с использованием оборудования ЦКП «Информатика» [19] эксперименты с тестами из пакета NPВ [20] показали, что границами по

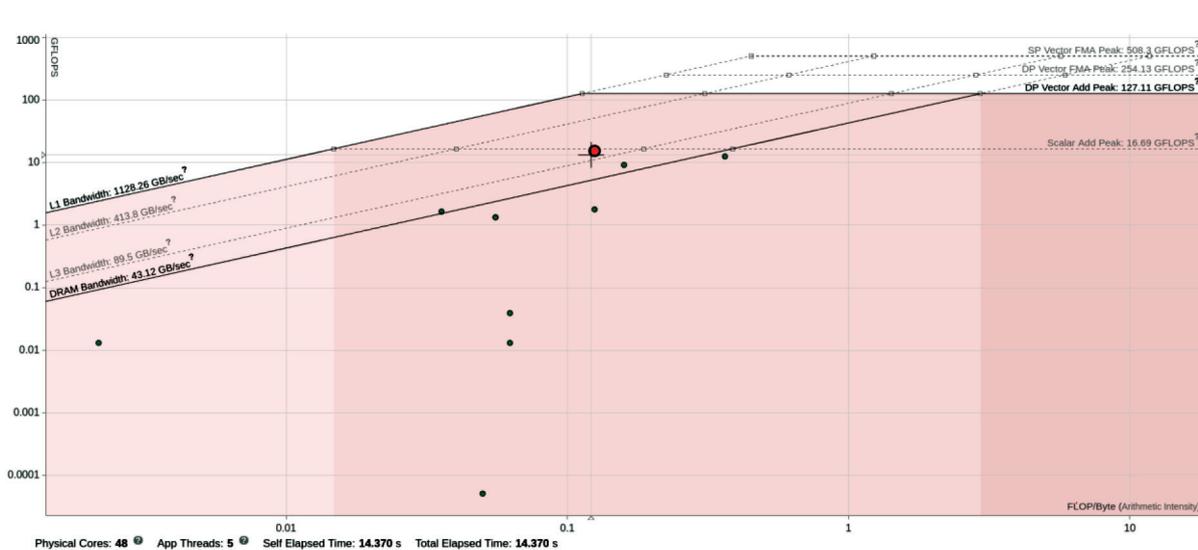


Рис. 1. Производительность приложений преобразования Фурье  
Fig. 1. Performance of Fourier transform applications

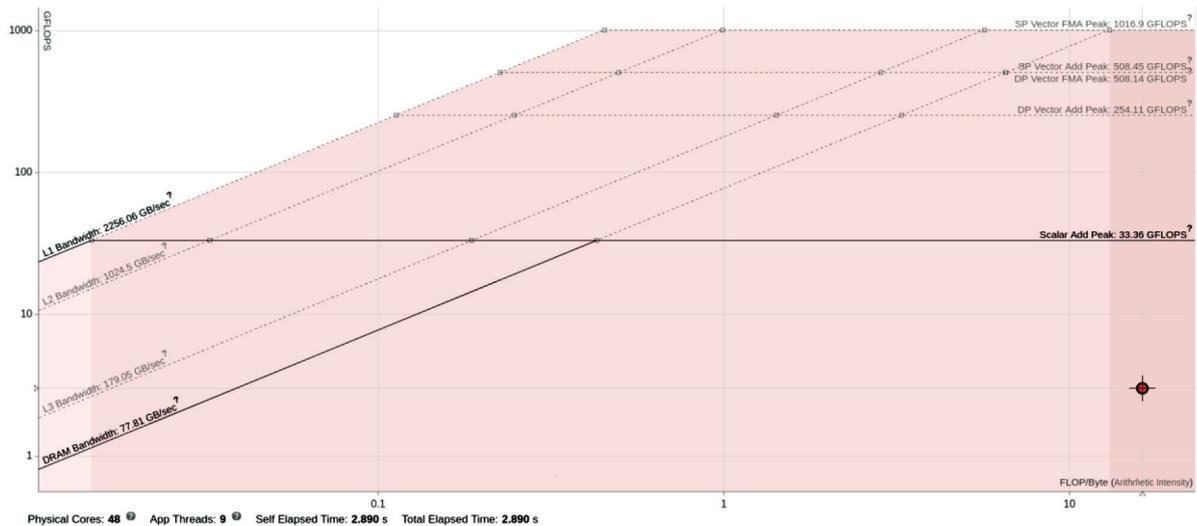


Рис. 2. Производительность приложений сортировки  
Fig. 2. Sorting application performance

производительности приложения могут служить как оперативная память, так и вычислительные блоки.

На рис. 1 представлено графическое изображение Roofline-модели приложения с использованием операций преобразования Фурье при выполнении на платформе Intel Xeon.

Из рис. 1 видно, что ограничивающими факторами для основного цикла приложения являются производительность процессора по скалярным операциям (*Scalar Add Peak*) и производительность кеша памяти второго уровня.

На основе Roofline-модели можно определить пути оптимизации приложений. В рассмотренном на рис. 1 примере направлением оптимизации является переход к векторным технологиям архитектуры Intel Xeon и переструктуризация приложения таким образом, чтобы задействовать функционирование кеша первого уровня. В этом случае основной цикл программы может быть сдвинут вертикально вверх по диаграмме, что будет означать более эффективное использование возможностей вычислительной архитектуры по производительности на операциях с плавающей точкой.

Другим направлением оптимизации может быть уменьшение соотношения операций с плавающей точкой и объема обмена с памятью. При этом основной цикл программы будет сдвинут на диаграмме вправо. В этом случае переход к векторным технологиям (движение вверх по диаграмме) может осуществляться без использования кеширования памяти.

На рис. 2 представлена Roofline-модель для приложений сортировки со случайным доступом к памяти.

Из рис. 2 видно, что главный цикл приложения находится в области, в которой соотношение между производительностью памяти и вычислительного

блока таково, что достаточно использования RAM без кеширования для любой технологии вычислений — скалярной или векторной. При этом использование приложением производительности вычислительных блоков не является оптимальным. Имеется многократный запас по производительности скалярной технологии и еще более существенный для векторной технологии.

### Заключение

Современные тенденции предоставления информационных сервисов как облачных услуг цифровых платформ требуют выработки новых подходов к организации вычислительного процесса суперкомпьютеров и гибридных вычислительных комплексов.

Отличительной особенностью цифровых платформ для научных исследований является аккумуляция разнородных научных задач из разных областей прикладной и фундаментальной науки, а также междисциплинарных задач. Задачи, требующие вычислительных сервисов и ресурсов размещаются средствами цифровой платформы в разного рода вычислительных комплексах, зарегистрированных как сервисы платформы.

В этих условиях организация вычислительного процесса в гибридных высокопроизводительных комплексах должна быть направлена с одной стороны на обеспечение возможности одновременного исполнения разнотипных задач, часто требующих создания вычислительных сред, не совместимых между собой. С другой стороны должны разрабатываться методы по оптимизации загрузки вычислительного комплекса, позволяющий максимально использовать возможности вычислительной инфраструктуры по выполнению математических расчетов.

Задачи первого направления решаются с использованием технологий виртуализации, применения облачных сервисов и средств цифровой платформы для научных исследований.

Решение задач второго направления возможно за счет применения предложенных в статье методов и подходов, которые позволяют определять априори требуемые задачами вычислительные ресурсы и на основе данной информации планировать загрузку и выполнение заданий гибридного высокопроизводительного вычислительного комплекса.

Для повышения эффективности выполнения вычислительных заданий в состав сервисов цифровой платформы и высокопроизводительных вычислительных комплексов должны входить средства анализа эффективности программного кода пользователей приложений.

Построение интуитивно понятных графических Roofline моделей позволит пользователям высокопроизводительных комплексов в составе цифровой платформы научных исследований определять «узкие места» в программном коде и выбирать пути оптимизации использования вычислительных ресурсов.

#### Библиографический список

1. Журавлев А. А., Ревизников Д. Л., Абгарян К. К. Метод дискретных элементов с атомарной структурой // Материалы XXI международной конференции по вычислительной механике и современным прикладным программным системам (ВМСППС'2019), Алушта, 24–31 мая 2019 г. М.: Московский авиационный институт, 2019. С. 59—61.
2. Карцев А., Мальковский С. И., Волович К. И., Сорочкин А. А. Исследование производительности и масштабируемости пакета Quantum ESPRESSO при изучении низкоразмерных систем на гибридных вычислительных системах // Математическое моделирование в материаловедении электронных компонентов: материалы I международной конференции, Москва, 21–23 октября 2019 г. М.: МАКС Пресс, 2019. С. 18—20. DOI: 10.29003/m682.MMMSEC-2019
3. Vouzis P. D., Sahinidis N. V. GPU-BLAST: using graphics processors to accelerate protein sequence alignment // *Bioinformatics*. 2011. V. 27, Iss. 2. P. 182—188. DOI: 10.1093/bioinformatics/btq644
4. Микурова А. В., Скворцов В. С. Создание обобщенной модели предсказания ингибирования нейраминидазы вируса гриппа различных штаммов // *Биомедицинская химия*. 2018. Т. 64, № 3. С. 247—252. DOI 10.18097/PBMC20186403247
5. Микурова А. В., Скворцов В. С., Раевский О. А. Компьютерная оценка селективности ингибирования мускариновых рецепторов M1–M4 // *Biomedical Chemistry: Research and Methods*. 2018. Т. 1, № 3. С. e00072. DOI: 10.18097/BMCRM00072
6. Горчаков А. Ю., Посыпкин М. А. Сравнение вариантов многопоточной реализации метода ветвей и границ для многоядерных систем // *Современные информационные технологии и ИТ-образование*. 2018. Т. 14, № 1. С. 138—148. DOI: 10.25559/STITO.14.201801.138–148
7. Berriman G. B., Deelman E., Juve G., Rynge M., Vöckler J.–S. The application of cloud computing to scientific workflows: a study of cost and performance // *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*. 2013. V. 371, Iss. 1983. P. 20120066. DOI: 10.1098/rsta.2012.0066
8. Якобовский М. В., Бондаренко А. А., Выродов А. В., Григорьев С. К., Корнилина М. А., Плотников А. И., Поляков С. В., Попов И. В., Пузырьков Д. В., Суков С. А. Облачный сервис для решения многомасштабных задач нанотехнологии на кластерах и суперкомпьютерах // *Известия ЮФУ. Технические науки*. 2016. № 12. С. 103—114. DOI: 10.18522/2311-3103-2016-12-103114
9. Ding F., an Mey D., Wienke S., Zhang R., Li L. A study on today's cloud environments for HPC applications // In: Helfert M., Desprez F., Ferguson D., Leymann F. (eds) *Cloud Computing and Services Science. CLOSER 2013. Communications in Computer and Information Science*. V. 453. Cham: Springer, 2014. P. 114—127. DOI: 10.1007/978-3-319-11561-0\_8
10. Волович К. И., Зацаринный А. А., Кондрашев В. А., Шабанов А. П. О некоторых подходах к представлению научных исследований как облачного сервиса // *Системы и средства информатики*. 2017. Т. 27, № 1. С. 73—84. DOI: 10.14357/08696527170105
11. Абрамов С. М. 2018: анализ суперкомпьютерных кибераинфраструктур ведущих стран мира // Суперкомпьютерные технологии (СКТ–2018): материалы 5-й Всероссийской научно-технической конференции, Дивноморское, Геленджик, 17–22 сентября 2018 г. Т. 1. Дивноморское, Геленджик: Южный федеральный университет, 2018. С. 11—18.
12. Абрамов С. М., Лилитко Е. П. Состояние и перспективы развития вычислительных систем сверхвысокой производительности // *Информационные технологии и вычислительные системы*. 2013. № 2. С. 6—22.
13. Клинов М. С., Лапшина С. Ю., Телегин П. Н., Шабанов Б. М. Особенности использования многоядерных процессоров в научных вычислениях // *Вестник Уфимского государственного авиационного технического университета*. 2012. Т. 16, № 6. С. 25—31.
14. Абрамов С. М. Правда, искажающая истину. Как следует анализировать Top500? // *Вестник Южно-уральского государственного университета*. Серия: Вычислительная математика и информатика. 2013. Т. 2, № 3. С. 5—31. DOI: 10.14529/cmse130301
15. Afanasyev I., Voevodin V. The comparison of large-scale graph processing algorithms implementation methods for Intel KNL and NVIDIA GPU // In: Voevodin V., Sobolev S. (eds) *Supercomputing. RuSCDays 2017. Communications in Computer and Information Science*. V. 793. Cham: Springer, 2017. P. 80—94. DOI: 10.1007/978-3-319-71255-0\_7
16. Sobolev S. I., Antonov A. S., Shvets P. A., Nikitenko D. A., Stefanov K. S., Voevodin Vad. V., Voevodin Vl. V., Zhumatiy S. A. Evaluation of the octotron system on the lomonosov-2 supercomputer // *Параллельные вычислительные технологии (ПаВТ'2018): труды международной научной конференции, Ростов-на-Дону, 2–6 апреля 2018 г. Челябинск: Издательский центр ЮУрГУ*, 2018. С. 176—184.
17. Zatsarinny A. A., Gorshenin A. K., Kondrashev V. A., Volovich K. I., Denisov S. A. Toward high performance solutions as services of research digital platform // *Procedia Computer Science: Proc. 13th International Symposium "Intelligent Systems", INTELS 2018, St. Petersburg, 22–24 October 2018*. V. 150. St. Petersburg: Elsevier B.V., 2019. P. 622—627. DOI: 10.1016/j.procs.2019.02.078
18. Williams S., Waterman A., Patterson D. Roofline: an insightful visual performance model for multicore architectures // *Communications of the ACM*. 2009. V. 52, N 4. P. 65—76. DOI: 10.1145/1498765.1498785
19. Положение о Центре коллективного пользования «Информатика». URL: <https://www.frccsc.ru/ckp> (дата обращения: 15.01.2020).
20. NAS Parallel Benchmarks (NPB). URL: <https://www.nas.nasa.gov/publications/npb.html> (дата обращения: 15.01.2020).

Эксперименты по измерению производительности приложений при предоставлении облачных сервисов проведены с использованием вычислительных ресурсов ЦКП «Информатика» ФИЦ ИУ РАН.

Статья поступила в редакцию 11 декабря 2020 г.

*Izvestiya vuzov. Materialy elektronnoi tekhniki = Materials of Electronics Engineering.* 2020, vol. 23, no. 4, pp. 289—296.  
DOI: 10.17073/1609-3577-2020-4-289-296

## Estimation of the workload of a hybrid computing cluster when performing modeling tasks in materials science

K. I. Volovich<sup>1,§</sup>

<sup>1</sup> *Federal Research Center “Computer Science and Control” of the Russian Academy of Sciences, 44–2 Vavilov Str., Moscow 119333, Russia*

**Abstract.** The article is devoted to methods of calculation and evaluation of the efficiency of functioning of hybrid computing systems. Material science software systems demonstrate maximum efficiency when operating on hybrid computing systems when using graphics accelerators for calculations. Examples include the VASP (The Vienna Ab initio Simulation Package) and Quantum ESPRESSO software systems. These software systems are most efficient when using monopolistic computing resources: RAM, CPU, GPU.

When operating a hybrid high-performance cluster, the problem arises of resource management and their division between a group of users. Technologies need to be developed that ensure the allocation of resources to materials science applications for different users and research teams. The modern approach to organizing the computing process is the use of virtualization and cloud technologies. Cloud technologies enable the provision of SaaS and PaaS services to users. It is advisable to provide scientific teams with applied materials science systems as cloud services.

Such diverse approaches, when applied in a single computer complex, require the development of methods for optimizing the load on the resources of a high-performance complex, assessing the efficiency of using its computational capabilities, and developing methods for improving user programs.

Determining the quality of the complex loading is an important task when providing high-performance computing services to research teams performing interdisciplinary research in various fields of science and technology. The article proposes a method for calculating the value of the load value using the peak performance values of the complex. The results and performance quality of high performance computing cloud scientific services are analyzed using a Roofline model.

**Keywords:** hybrid architecture, efficiency, code optimization, cloud service, graphics accelerator, load estimation, efficiency assessment

### References

- Zhuravlev A. A., Reviznikov D. L., Abgaryan K. K. The method of discrete elements with an atomic structure. *Proc. XXI International Conference on Computational Mechanics and Modern Applied Software Systems. Alushta, May 24–31, 2019.* Moscow: Moskovskiy aviatsionnyy institut, 2019. Pp. 59—61. (In Russ.)
- Kartsev A., Malkovsky S. I., Volovich K. I., Sorokin A. A. Investigation of performance and scalability of the Quantum ESPRESSO package in the study of low-dimensional systems on hybrid computing systems. *Mathematical Modeling in Materials Science of Electronic Component: Proc. I International Conference. Moscow, October 21–23, 2019.* Moscow: MAKS Press, 2019. Pp. 18—20. (In Russ.) DOI: 10.29003/m682.MMMSEC-2019
- Vouzis P. D., Sahinidis N. V. GPU-BLAST: using graphics processors to accelerate protein sequence alignment. *Bioinformatics.* 2011, vol. 27, iss. 2, pp. 182—188. DOI: 10.1093/bioinformatics/btq644
- Mikurova A. V., Skvortsov V. S. A generalized prediction model of inhibition of neuraminidase of influenza virus of various strains. *Biochemistry (Moscow), Supplement Series B: Biomedical Chemistry.* 2018, vol. 12, no. 4, pp. 322—329. DOI: 10.1134/S1990750818040054
- Mikurova A. V., Skvortsov V. S., Raevsky O. A. Computational evaluation of selectivity of inhibition of muscarinic receptors M1–M4. *Biomedical Chemistry: Research and Methods.* 2018, vol. 1, no. 3, p. e00072. (In Russ.) DOI: 10.18097/BMCRM00072
- Gorchakov A. Ju., Posypkin Comparison of variants of multithreading realization of method of branches and borders for multi-core systems. *Modern Information Technologies and IT-Education.* 2018, vol. 14, no. 1, pp. 138—148. (In Russ.) DOI: 10.25559/SITITO.14.201801.138-148
- Berriman G. B., Deelman E., Juve G., Rynge M., Vöckler J.–S. The application of cloud computing to scientific workflows: a study of cost and performance. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences.* 2013, vol. 371, iss. 1983, p. 20120066. DOI: 10.1098/rsta.2012.0066
- Yakovovskiy M. V., Bondarenko A. A., Vyrodov A. V., Grigoriev S. K., Kornilina M. A., Plotnikov A. I., Polyakov S. V., Popov I. V., Puzyrkov D. V., Soukov S. A. Cloud service for solution of multiscale nanotechnology problems on clusters and supercomputers. *Izvestiya SFedU. Engineering Sciences.* 2016, no. 12, pp. 103—114. (In Russ.) DOI: 10.18522/2311-3103-2016-12-103114
- Ding F., an Mey D., Wienke S., Zhang R., Li L. A study on today's cloud environments for HPC applications. In: Helfert M., Desprez F., Ferguson D., Leymann F. (eds) *Cloud Computing and Services Science. CLOSER 2013. Communications in Computer and Information Science, vol. 453.* Cham: Springer, 2014. Pp. 114—127. DOI: 10.1007/978-3-319-11561-0\_8
- Volovich K. I., Zatsarinnyy A. A., Kondrashev V. A., Shabanov A. P. Scientific research as a cloud service. *Systems and*

### Information about author:

**Konstantin I. Volovich<sup>1,§</sup>** — Cand. Sci. (Eng.), Senior Researcher, e-mail: kvolovich@frcsc.ru

§ Corresponding author

*Means of Informatics*. 2017, vol. 27, iss. 1, pp. 73—84. (In Russ.). DOI: 10.14357/08696527170105

11. Abramov S. M. 2018: Analysis of supercomputer cyber infrastructures of the leading countries of the world. In: *Supercomputer Technologies: Proc. 5th All-Russian scientific and technical conference. Divnomorskoye, Gelendzhik, September 17–22, 2018, vol. 1*. Divnomorskoye, Gelendzhik: Yuzhnyy federal'nyy universitet, 2018. Pp. 11—18. (In Russ.)

12. Abramov S. M., Lilitko E. P. Current state and development prospects of high-end HPC systems. *Journal of Information Technologies and Computing Systems*. 2013, no. 2, pp. 6—22. (In Russ.)

13. Klinov M. S., Lapshina S. Yu., Telegin P. N., Shabanov B. M. Multicore processing features in scientific computing. *Vestnik UGATU*. 2012, vol. 16, no. 6, pp. 25—31. (In Russ.)

14. Abramov S. M. True judgments that distort the real truth. How to analyze the Top500? *Bulletin of the South Ural State University Computational Mathematics and Software Engineering*. 2013, vol. 2, no. 3, pp. 5—31. (In Russ.). DOI: 10.14529/cmse130301

15. Afanasyev I., Voevodin V. The comparison of large-scale graph processing algorithms implementation methods for Intel KNL and NVIDIA GPU. In: Voevodin V., Sobolev S. (eds) *Supercomputing. RuSCDays 2017. Communications in Computer and Information Science, vol. 793*. Cham: Springer, 2017. Pp. 80—94. DOI: 10.1007/978-3-319-71255-0\_7

16. Sobolev S. I., Antonov A. S., Shvets P. A., Nikitenko D. A., Stefanov K. S., Voevodin Vad. V., Voevodin Vl. V., Zhumatiy S. A. Evaluation of the octotron system on the lomonosov-2 supercomputer. *Parallel Computing Technologies (PaVT'2018): Proc. International*

*Scientific Conference. Rostov-on-Don, April 2–6, 2018*. Chelyabinsk: Izdatel'skiy tsentr YUUrGU, 2018. Pp. 176—184.

17. Zatsarinny A. A., Gorshenin A. K., Kondrashev V. A., Volovich K. I., Denisov S. A. Toward high performance solutions as services of research digital platform. *Procedia Computer Science: Proc. 13th International Symposium «Intelligent Systems», INTELS 2018. St. Petersburg, 22–24 October 2018, vol. 150*. St. Petersburg: Elsevier B.V., 2019. Pp. 622—627. DOI: 10.1016/j.procs.2019.02.078

18. Williams S., Waterman A., Patterson D. Roofline: an insightful visual performance model for multicore architectures. *Communications of the ACM*. 2009, vol. 52, no. 4, pp. 65—76. DOI: 10.1145/1498765.1498785

19. Regulations on the Center for Collective Use “Informatics”. (In Russ.). URL: <https://www.frccsc.ru/ckp> (accessed: 15.01.2020).

20. NAS Parallel Benchmarks (NPB). URL: <https://www.nas.nasa.gov/publications/npb.html> (accessed: 15.01.2020).

### Acknowledgments

*This work was supported in part by the Russian Foundation for Basic Research (projects No. 18–29–03100 and No. 18–07–00669).*

*Experiments to measure the performance of applications in the provision of cloud services were carried out using the computing resources of the Center for Collective Use “Informatics” of the Federal Research Center “Computer Science and Control” of the Russian Academy of Sciences.*

Received December 11, 2020