

Вычисление интегралов в теории функционала электронной плотности в приближении электронного газа с использованием технологии CUDA

А. А. Сокуров

Кабардино-Балкарский научный центр Российской академии наук,
Российская Федерация, 360000, Нальчик, ул. Шортанова, 89А

Для цитирования: Сокуров А. А. Вычисление интегралов в теории функционала электронной плотности в приближении электронного газа с использованием технологии CUDA // Вестник Санкт-Петербургского университета. Прикладная математика. Информатика. Процессы управления. 2024. Т. 20. Вып. 3. С. 335–349.
<https://doi.org/10.21638/spbu10.2024.303>

Выражение для потенциальной энергии взаимодействия двух нейтральных атомов в отсутствие химической связи состоит из суммы кратных и несобственных интегралов. Ввиду громоздкости функций нахождение этих интегралов в явном виде не представляется возможным. Широко используемые на практике программные комплексы на базе стандартных методов вычислительной математики также не способны за короткое время обеспечить удовлетворительную точность при их численном расчете. В квантовой химии и вычислительной физике сказанное сильно ограничивает подходы к моделированию свойств и структур атомных/молекулярных систем. Приходится полагаться на не столь результативные в плане точности методы интегрирования Монте-Карло или формулы типа Гаусса — Лагерра. В настоящей статье применительно к данной проблеме предлагается методика переноса квадратурных формул Ньютона — Котеса на архитектуру графических процессоров. Подробно обсуждаются особенности такого переноса, призванные устранить тонкие места и максимально повысить производительность соответствующих компьютерных программ. В качестве платформы для массивно-параллельных вычислений выступает технология CUDA от компании NVIDIA. Тестирование показало, что в типичных задачах эффективность программ для графических процессоров, основанных на распараллеленных аналогах, в среднем на порядок выше классических. В рамках предложенного подхода удалось с высокой точностью и за приемлемое машинное время рассчитать межатомные потенциалы взаимодействия в широком диапазоне изменения расстояний между атомами, а также определить равновесные параметры взаимодействия. Полученные результаты хорошо согласуются с известными из литературы данными.

Ключевые слова: параллельное программирование, численное интегрирование, технология CUDA, GPGPU, метод функционала электронной плотности.

1. Введение. Приближенное вычисление интегралов является одной из классических и фундаментальных проблем в вычислительной математике. К настоящему времени разработано большое количество методов для численного расчета определенных интегралов, обладающих различной степенью точности и сложности [1], а также основанных на них высокоэффективных программных комплексов. Однако, несмотря на относительно давнюю историю и хорошую освещенность данной темы в научной литературе, особого внимания на практике продолжает требовать задача численного интегрирования по многомерным областям. С целью раскрыть суть проблемы прове-

дем сравнительный анализ вычислительной сложности алгоритмов, предназначенных для интегрирования функций от одной и двух переменных.

Рассмотрим некоторую квадратурную формулу второго порядка точности для одномерного интеграла. Если известен главный член погрешности, то будет справедлива оценка [2]

$$\varepsilon = \left| I - \tilde{I} \right| \approx Mh^2, \quad (1)$$

где I и \tilde{I} — точное и приближенное значения искомого интеграла соответственно; h — шаг равномерной сетки; M — независимая от шага постоянная. Предположим далее, что квадратурную формулу путем последовательного интегрирования можно применить к вычислению двойного (кратного) интеграла и что подобное обобщение также имеет второй порядок точности относительно шагов двумерной сетки h_x и h_y . Для семейства методов Ньютона — Котеса все сказанное, как правило, выполняется. Тогда по аналогии с (1) запишем, что

$$\varepsilon = \left| I - \tilde{I} \right| \approx M_1 h_x^2 + M_2 h_y^2. \quad (2)$$

Из сравнения правых частей (1) и (2) следует важный вывод: величина погрешности ε в (2) будет одинакового порядка малости с ε в (1) только, если шаги по h_x и h_y будут одинакового порядка малости с h . Другими словами, расчетная сетка из 10^k точек для одномерного интеграла в «среднем» эквивалентна сетке из 10^{2k} точек для двойного, 10^{3k} для тройного и т. д. Таким образом, достижение сопоставимой точности в случае многомерного интеграла потребует на порядки большего числа расчетных точек. Как итог — на порядки увеличиваются и расходы вычислительных ресурсов, в первую очередь процессорного времени. По этой причине на практике зачастую используют различные варианты не столь ресурсоемкого, но в то же время и не столь эффективного метода Монте-Карло.

Необходимость в численном расчете многомерных интегралов очень часто возникает во многих разделах физики. Типичными примерами служат выражения, определяющие потенциалы в электростатике, магнитостатике, теории гравитации, теории межатомного взаимодействия, электродинамике и квантовой механике. Встречаются они и при решении краевых задач математической физики (например, для уравнений теплопроводности, Лапласа, Пуассона).

Ввиду многочисленных приложений разработка экономичных численных методов для вычисления многомерных интегралов остается чрезвычайно актуальным направлением в теории вычислительных методов. Помимо классических подходов в современной литературе регулярно появляются статьи, посвященные данному вопросу. Среди них можно выделить [3], в которой на кратный случай обобщен известный метод Ромберга, традиционно применяемый для повышения точности расчета одномерных интегралов и реализованный практически во всех математических пакетах. В [4] представлены формулы для точного вычисления погрешности аппроксимации кратных стохастических интегралов Ито. Наблюдается тенденция использования в подходах систем компьютерной алгебры [5, 6]. Численные методы интегрирования двойных интегралов с переменными границами предложены в [7]. В свете обсуждаемой нами физической задачи выделим еще статьи [8, 9]. В них авторами разработаны методы и модификации квадратурных формул для расчета электростатического потенциала многоатомной системы и потенциала простого слоя. Стоит также отметить использование ЛПТ последовательности двоично-рациональных псевдослучайных чисел в ме-

тоде Монте-Карло, позволяющего в ряде случаев значительно повысить эффективность численных расчетов многомерных интегралов [10].

Список работ, затрагивающих тему численного интегрирования, довольно обширный. Однако приходится констатировать, что абсолютное их большинство посвящено методам расчета интегралов специальных видов, возникающих при решении тех или иных прикладных задач. Ускорения сходимости удается добиться, принимая во внимание особенности поведения подынтегральных функций. По-видимому, в общем случае существенно уменьшить вычислительную сложность алгоритмов, обусловленную размерностью два и выше, принципиально невозможно.

В связи с таким положением дел едва ли не единственным выходом из сложившейся ситуации видится более широкое внедрение в численный анализ идей параллелизма. Параллельные вычисления в течение последних десятилетий активно и с успехом применяются в физике при моделировании нелинейных динамических систем [11–13], в линейной алгебре для ускорения векторных и матричных операций, в вычислительной математике при программировании разностных алгоритмов, а также в области обработки больших объемов данных, машинного обучения и искусственного интеллекта. В немалой степени этому способствует распространенность компьютерной техники, ориентированной на выполнение распределенных операций.

В настоящей работе проводится численное моделирование межатомного взаимодействия двух нейтральных атомов в рамках метода функционала электронной плотности (DFT — Density Functional Theory). Как известно, химическому связыванию атомов предшествует действие дисперсионных сил притяжения. В работе рассматривается именно данный тип взаимодействия при достаточно больших расстояниях между атомами, т. е. когда это расстояние превышает сумму ковалентных радиусов атомов. Составляющие потенциала взаимодействия с учетом хартри-фоковских орбиталей — кулоновская и квантово-статистическая — записываются в форме несобственных кратных интегралов, которые в терминах элементарных функций явно не выражаются. Вычисление интегралов подобного типа вызывает значительные затруднения при моделировании атомных систем в рамках квантовой химии и вычислительной физики. На практике и в научных исследованиях для расчета структуры и свойств атомных/молекулярных систем обычно используют такие программные пакеты как Gaussian, ORCA, GAMESS и NWChem. К большому сожалению, ни одна из этих программ и им аналогичных не располагает возможностями, позволяющими табулировать межатомный потенциал в широком диапазоне изменения расстояний между частицами. Для решения этой проблемы с высокой точностью в работе разработан и программно реализован алгоритм распараллеливания методов Ньютона — Котеса, адаптированных под двойной интеграл по прямоугольной области. В его основе лежит технология массивно-параллельных вычислений CUDA от компании NVIDIA. Алгоритм тестируется применительно к потенциалу невалентного взаимодействия двух нейтральных атомов аргона, кремния и углерода. Полученные результаты сравниваются с известными из литературы данными.

Методика расчетов излагается в контексте решения прикладной задачи, но она не накладывает каких-либо ограничений на вид функций и может быть с успехом использована для численного интегрирования произвольно заданных функций.

2. Математическая модель. В данной части статьи дается описание формализованной математической модели взаимодействия пары атомов, выписываются интегральные выражения для составляющих потенциала и выводятся необходимые в дальнейшем количественные оценки.

В приближении электронного газа физическая теория парного нековалентного (т.е. без образования химических связей) взаимодействия атомов была разработана в работах [14, 15]. В ней предполагается, что результирующая электронная плотность ρ_{tot} есть сумма электронных плотностей ρ_a и ρ_b каждого из атомов:

$$\rho_{tot} = \rho_a + \rho_b.$$

При записи всех формул по тексту используется атомная система единиц

$$\hbar = e = m_e = 1,$$

в которой \hbar — постоянная Планка, e , m_e — заряд и масса электрона. Электронная плотность ρ отдельного атома с порядковым номером z в периодической таблице находится по формуле [16]

$$\rho(r) = \frac{z}{4\pi r} \left[\sum_{i=1}^A \gamma_i^A \lambda_i^2 e^{-\lambda_i r} + \sum_{j=1}^B \gamma_j^B \lambda_j ({}^B\gamma_j^B \lambda_j - 2) e^{-{}^B\lambda_j r} \right],$$

где ${}^A\gamma_i$, ${}^A\lambda_i$, ${}^B\gamma_j$, ${}^B\lambda_j$ — числовые данные; r — расстояние от ядра. Верхние пределы в знаках суммы опущены, так как количество слагаемых в них варьируется от номера к номеру. Данная формула достаточно корректно описывает оболочечную структуру атома в рамках метода Хартри — Фока и выражает вероятность обнаружения электрона на расстоянии r от ядра. Потенциальная энергия U взаимодействия двух атомов зависит от расстояния R между их ядрами и состоит из кулоновского U_C и квантово-статистического U_G вкладов: $U(R) = U_C(R) + U_G(R)$. Кулоновскую часть потенциальной энергии можно представить в виде интеграла

$$U_C(R) = 16\pi^2 \int_0^{+\infty} \int_0^{+\infty} \rho_a(r_1) \rho_b(r_2) I(r_1, r_2; R) r_1^2 r_2^2 dr_1 dr_2. \quad (3)$$

Для гомоатомных и гетероатомных пар функция I записывается по-разному. Если пара гомоатомная, то

$$I(r_1, r_2; R) = \frac{1}{R} + F(r_1, r_2; R) - \frac{4}{R + r_1 + |R - r_1|},$$

если гетероатомная —

$$I(r_1, r_2; R) = \frac{1}{R} + F(r_1, r_2; R) - \frac{2}{R + r_1 + |R - r_1|} - \frac{2}{R + r_2 + |R - r_2|}.$$

Определение F , входящей в I , следующее:

$$F(r_1, r_2; R) = \begin{cases} \frac{2}{R + r_1 + |R - r_1|}, & r_2 \leq |R - r_1|, \\ \frac{1}{2} \left(\frac{1}{r_1} + \frac{1}{r_2} \right) - \frac{R}{4r_1 r_2} - \frac{(r_1 - r_2)^2}{4Rr_1 r_2}, & |R - r_1| < r_2 < R + r_2, \\ \frac{1}{r_2}, & r_2 \geq R + r_1. \end{cases}$$

Интеграл (3) — кратный и несобственный, а его вычисление представляет собой нетривиальную задачу. Обычно интегралы типа (3) принято рассчитывать методом Гаусса — Лагерра [17]. В рассматриваемом случае, однако, метод оказывается

не столь результативным, несмотря на показательные расчеты из [14]. Связано это с тем, что подынтегральная функция для атомных систем имеет нерегулярный осциллирующий характер, вследствие чего достижение приемлемой точности требует вычисления корней полиномов Лагерра порядка ста. Во-вторых, большая часть корней приходится на участки, где значения функций ничтожно малы и не приносят ощутимый вклад в сумму. Наконец, при таком подходе становится проблематичным контролировать точность результата. При всем богатстве встроенного функционала совершенно непригодными здесь оказываются и многие пакеты компьютерной математики, например Mathcad, Maple и Wolfram Mathematica.

Для эффективной табуляции (3) по параметру R предлагается использовать параллельные модификации распространенных квадратурных формул. Но прежде придется перейти от несобственного интеграла к собственному, выяснив, насколько далеко простираются «главные» значения подынтегральной функции. Хотя в атомной физике расстояния порядка нескольких десятков атомных единиц длины уже являются хорошим приближением к бесконечности (силы Ван-дер-Ваальса убывают с расстоянием R по степенному закону $U(R) \sim R^{-6}$), вопросу можно придать количественную форму.

Заменим в (3) область интегрирования квадратом со стороной L . Такая замена для фиксированной атомной пары приводит к ошибке ε , которая зависит от L и R :

$$\varepsilon_C(R, L) = \left| U_C(R) - \tilde{U}_C(R, L) \right| = \left| \int_0^{+\infty} \int_0^{+\infty} f(r_1, r_2; R) dr_1 dr_2 - \int_0^L \int_0^L f(r_1, r_2; R) dr_1 dr_2 \right|.$$

Полагаясь на свойство аддитивности, имеем погрешность

$$\varepsilon_C < J_1 + J_2 + J_3, \tag{4}$$

где

$$J_1 = \int_L^{+\infty} \int_L^{+\infty} |f(\dots)| dr_1 dr_2, \quad J_2 = \int_0^L \int_L^{+\infty} |f(\dots)| dr_1 dr_2, \quad J_3 = \int_L^{+\infty} \int_0^L |f(\dots)| dr_1 dr_2.$$

Ограничимся здесь случаем гомоатомных пар. После несложных преобразований и оценок подынтегральной функции в пределах соответствующих областей для J_k найдем, что

$$J_1 < z^2 \left(\frac{1}{R} + \frac{3}{L} \right) \times \left(\sum_{i=1}^A \gamma_i e^{-A \lambda_i L} (1 + A \lambda_i L) + \sum_{j=1}^B \gamma_j^B \lambda_j L^2 e^{-B \lambda_j L} \right)^2, \tag{5}$$

$$J_2 < z^2 \left(\frac{1}{R} + \frac{3}{L} \right) \times \left(\sum_{i=1}^A \gamma_i e^{-A \lambda_i L} (1 + A \lambda_i L) - \sum_{j=1}^B \gamma_j^B \lambda_j L^2 e^{-B \lambda_j L} \right) \times \left(\sum_{i=1}^A \gamma_i \left[1 - e^{-A \lambda_i L} (1 + A \lambda_i L) \right] + \sum_{j=1}^B \gamma_j^B \lambda_j L^2 e^{-B \lambda_j L} \right), \tag{6}$$

$$J_3 < z^2 \left(\frac{1}{R} + \frac{1}{L} + \frac{4}{R+L} \right) \times \left(\sum_{i=1}^A \gamma_i e^{-A\lambda_i L} (1 + {}^B\lambda_i L) - \sum_{j=1}^B \gamma_j {}^B\lambda_j L^2 e^{-B\lambda_j L} \right) \times \\ \times \left(\sum_{i=1}^A \gamma_i \left[1 - e^{-A\lambda_i L} (1 + {}^A\lambda_i L) \right] + \sum_{j=1}^B \gamma_j {}^B\lambda_j L^2 e^{-B\lambda_j L} \right). \quad (7)$$

Неравенства типа (5)–(7) для гетероатомных пар не приводятся ввиду их чрезмерной громоздкости. Они получаются в ходе тех же рассуждений с учетом различия в определении функции I . Более того, в квантовой химии и методах молекулярной динамики энергию взаимодействия гетерогенной пары « a – b » принято выражать через энергии взаимодействия гомоатомных пар « a – a » и « b – b » с помощью эмпирических правил комбинирования, например правила Лоренца – Бергло [18]. Поэтому оценок (5)–(7) может быть достаточно.

В итоге в (4) для заданного R простым перебором находится такое значение L , при котором определенный интеграл \tilde{U} по квадрату $[0, L] \times [0, L]$ аппроксимирует несобственный (3) с погрешностью ε_C . Практически сделать это несложно, поскольку правые части (5)–(7) монотонно убывают вместе с ростом L .

Квантово-статистический вклад в потенциальную энергию дается интегралом

$$U_G(R) = \frac{\pi R^3}{4} \int_{-1}^1 \int_1^{+\infty} [\rho_{tot} E_G(\rho_{tot}) - \rho_a E_G(\rho_a) - \rho_b E_G(\rho_b)] (\lambda^2 - \mu^2) d\lambda d\mu, \quad (8)$$

здесь E_G – плотность энергии электронного газа, равная сумме плотностей энергии Хартри – Фока E_{HF} и корреляционной энергии E_{corr} :

$$E_G(\rho) = E_{HF}(\rho) + E_{corr}(\rho),$$

$$E_{HF}(\rho) = C_k \rho^{2/3} + C_e \rho^{1/3},$$

$$E_{corr}(\rho) =$$

$$= \begin{cases} 2.711816\rho^{1/2} - 0.706052\rho^{1/3} - 3.819822\rho^{2/3} - 1.319676\rho^{5/6}, & \rho < \rho_1, \\ -0.7062 - 0.00633 \ln(\rho), & \rho_1 \leq \rho \leq \rho_2, \\ -0.010367 \ln(\rho) - (0.001861 \ln(\rho) + 0.008869) \rho^{-1/3} - 0.062849, & \rho > \rho_2, \end{cases}$$

$$C_e = -\frac{3}{4} \left(\frac{3}{\pi} \right)^{1/3}, \quad C_k = \frac{3}{10} (3\pi^2)^{2/3}, \quad \rho_1 = 2.387324 \cdot 10^{-4}, \quad \rho_2 = 0.696013,$$

$$\rho_a(\lambda, \mu; R) = \rho_a \left(\frac{R}{2} (\lambda + \mu) \right), \quad \rho_b(\lambda, \mu; R) = \rho_b \left(\frac{R}{2} (\lambda - \mu) \right).$$

По аналогии с кулоновским вкладом оценим погрешность перехода в (8) к конечному пределу во внутреннем интеграле:

$$\varepsilon_G(R, L) = \left| U_G(R) - \tilde{U}_G(R, L) \right| = \\ = \left| \int_{-1}^1 \int_1^{+\infty} f(\lambda, \mu; R) d\lambda d\mu - \int_{-1}^1 \int_1^L f(\lambda, \mu; R) d\lambda d\mu \right| <$$

$$\begin{aligned}
&< \frac{2z(C_k + C_e + 3.633)}{\pi R^3} \left(\sum_{i=1}^4 4^A \gamma_i e^{-\frac{1}{2}^A \lambda_i R(1+L)} \left[1 - L + e^{A \lambda_i R(1+L)} \right] + \right. \\
&+ \sum_{j=1}^B \frac{B \gamma_j e^{-\frac{1}{2}^B \lambda_j R(1+L)}}{B \lambda_j} \left\{ e^{B \lambda_j R(1+L)} [4 + B \lambda_j R(L-1)] - \right. \\
&\quad \left. - (L-1) [4 + B \lambda_j R(L+1)] \right\} - \\
&\quad \left. - \frac{8^B \gamma_j e^{-\frac{1}{2}^B \lambda_j R(1+L)}}{B \lambda_j} \left[1 - L + e^{B \lambda_j R(1+L)} \right] \right).
\end{aligned}$$

Как и ранее, неравенство относится к гомоатомной паре.

Из оценок для «хвостов» интегралов следует, что меньшие значения R требуют больших значений L , поэтому проводить расчет приходится все дальше. Такой вывод находится в качественном согласии с поведением подынтегральных функций. К примеру, в интеграле кулоновского вклада функция I стремится к бесконечности при R , стремящемся к нулю, оказывая тем самым негативное влияние на сходимость. Однако область малых межатомных расстояний ($R \ll 1$) на практике, как правило, особого интереса не представляет. Межатомное взаимодействие оказывается настолько интенсивным, что в модельном потенциале на участке левее положения потенциальной ямы энергия взаимодействия тождественно полагается равной бесконечности (так называемый потенциал Сазерленда). Наибольший интерес представляют потенциальная яма и ее окрестность.

Наконец, когда в определенных интегралах \tilde{U}_C и \tilde{U}_G пределы интегрирования фиксированы должным образом, остается провести их численный расчет.

3. Параллелизация численных алгоритмов интегрирования. Как отмечалось в п. 1, все протестированные нами системы компьютерной математики не позволяют эффективно рассчитывать интегралы, возникающие в теории функционала электронной плотности. Процесс либо «зависает», либо выводятся сообщения об ошибках. Например, известный пакет Wolfram Mathematica выдает следующее предупреждение: «*NIntegrate: Numerical integration converging too slowly; suspect one of the following: singularity, value of the integration is 0, highly oscillatory integrand, or WorkingPrecision too small...*», указывая на медленную скорость сходимости из-за сильных осцилляций подынтегрального выражения. В таких условиях становится невозможным судить о корректности получаемых результатов. Вдобавок ситуация усугубляется требованием плотной табуляции потенциальной энергии в широком диапазоне изменения расстояния, а не только ее подсчетом в отдельно взятой точке.

В настоящей статье для решения этой ресурсоемкой задачи предлагается адаптировать под архитектуру графических процессоров параллельные варианты основных квадратурных формул — методы ячеек, трапеций, Симпсона и т. д. Свойство аддитивности обуславливает высокую степень распараллеливания процедуры интегрирования, благодаря чему время исполнения алгоритмов должно серьезно сократиться.

Сейчас существуют несколько программно-аппаратных платформ, позволяющих создавать массивно-параллельные приложения. Часть из них разработана под многоядерные или многопроцессорные системы, а другая под кластеры графических процессоров. Здесь будет использоваться CUDA (Compute Unified Device Architecture) — оригинальная архитектура для проведения вычислений с помощью графических карт от компании NVIDIA. Ниже, не вдаваясь в детали, опишем основные ее особенности [19, 20].

Технология CUDA превращает видеопроцессор (GPU) в программируемое наравне с центральным процессором (CPU) устройство и предназначена для решения отличных от рендеринга задач. Ее программная компонента реализована как расширение языка C. Стандартная CUDA-программа включает следующие этапы:

- выделение памяти на GPU;
- копирование данных из памяти CPU в память GPU;
- обработка данных на GPU;
- обратное копирование данных.

Ключевым понятием в CUDA является ядро — выполняемая на стороне GPU функция. Именно в ядре кодируется основная логика распараллеливания. Внутри GPU оно тиражируется множеством копий или нитей (по терминологии CUDA), инструкции которых затем исполняются многочисленными физическими ядрами GPU независимо. Для обеспечения кооперации нитей они объединяются в блоки, а блоки в сетку. Нити отличаются друг от друга значениями «вшитых» управляющей средой целочисленных переменных: *threadIdx* — индекс нити в блоке, *blockIdx* — индекс блока в сетке. Их комбинацией нитям ставятся в соответствие элементы в обрабатываемом массиве данных, если данные сами не генерируются по некоторому закону. Отдельно стоит выделить наличие разделяемой памяти *shared memory* — программируемой кэш-памяти, призванной избежать или свести к минимуму количество обращений в медленную оперативную память GPU.

Пусть необходимо вычислить интеграл I от функции $f(x, y)$ по прямоугольной области D . Введем в D сетку с шагами h_x и h_y . Приблизительно интеграл по элементарной ячейке $\Delta_{i,j}$ можно принять равным $S_{i,j} = RULE(\Delta_{i,j})$, где $RULE$ — произвольная квадратурная формула. Тогда

$$I \approx S = \sum_i \sum_j S_{i,j}. \quad (9)$$

Стремление добиться желаемой точности нередко приводит к очень плотной сетке, когда число слагаемых в (9) катастрофически велико. В таких случаях последовательное суммирование (9) мощностями только одного центрального процессора крайне затруднительно.

Чтобы произвести суммирование элементов матрицы $[S_{i,j}]$ параллельно средствами CUDA, сначала определимся с конфигурацией ядра: блок будет состоять из $n \times n$ нитей, сетка из $m_x \times m_y$ блоков, а блоку в разделяемой памяти выделяется массив *cache* под частичные суммы на n^2 элементов — по одному на нить. Методика подсчета (9) понятна из рис. 1.

Локальный индекс *threadIdx* нити в блоке транслируется в линейный индекс k в массиве *cache*. Глобальный же индекс нити в сетке, образованный парой *threadIdx* и *blockIdx*, транслируется в двумерный адрес (i, j) элемента в матрице S . После вычисления $S_{i,j}$ и сложения его с $cache_k$ индекс i получает приращение в общее число нитей в сетке по горизонтали. Оно хранится во встроенной переменной *gridDim*. Оказавшись по адресу (i', j) , нить, как и ранее, вычисленное $S_{(i',j)}$ складывает с $cache_k$. Смещение повторяется до тех пор, пока не будет зафиксирован выход за границы матрицы. Затем приращение получает второй индекс j , и описанная только что процедура запускается заново. В итоге части искомой величины S оказываются разбросанными по немногочисленным *cache*-ам всех блоков. Остается их скопировать в оперативную память CPU и там же просуммировать.

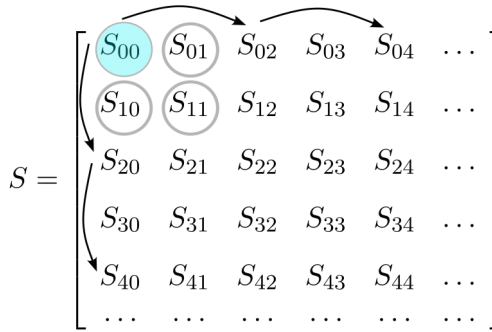


Рис. 1. Перемещение нити при обходе элементов матрицы
Кругами обозначена начальная расстановка.

Тестирование методики распараллеливания и сравнительный анализ эффективности расчетов на GPU и CPU проводились на основе метода ячеек. Разница в расчетном времени при решении тестовой задачи

$$I = \int_0^5 \int_0^5 \sin(e^x + e^y) dx dy$$

представлена на рис. 2, 3. Учитывались усредненные в ходе трех запусков программ показатели. Характеристики целевой системы были таковы:

- CPU — Intel Core i5-6400;
- GPU — NVIDIA GeForce GTX 1050;
- ОС — Windows 10 64-bit;
- версия CUDA — 11.8;
- среда разработки — MS Visual Studio 2022.

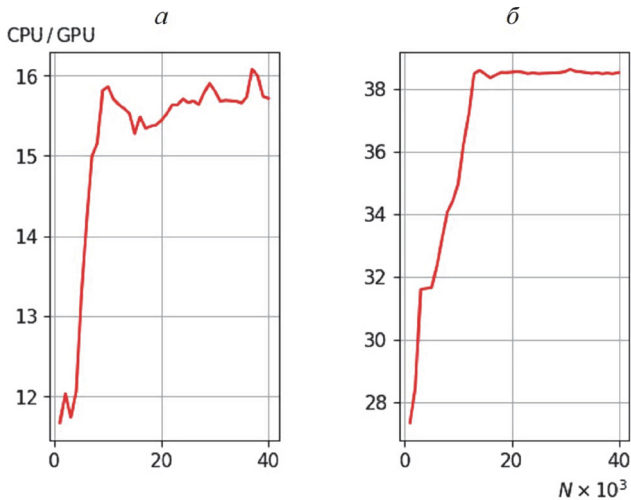


Рис. 2. Ускорение расчетов на GPU
 $N \times N$ — размерность сетки. а — тип *double*; б — тип *float*.

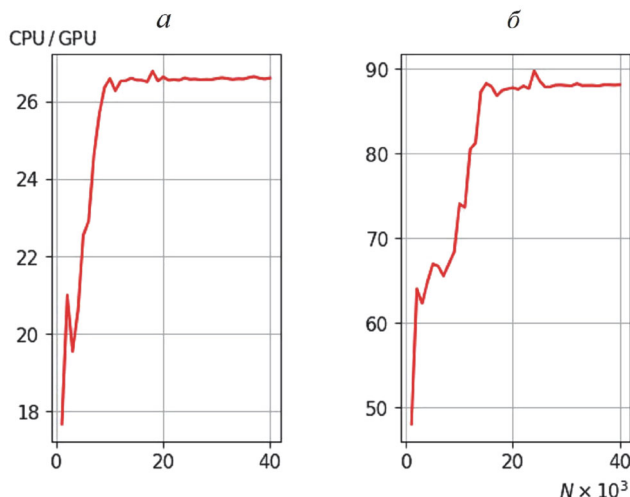


Рис. 3. Ускорение расчетов на GPU (оптимизация кода CPU отсутствует)
 $N \times N$ — размерность сетки. а — тип *double*; б — тип *float*.

Нужно отметить, что в тестах участвовала оптимизированная CPU-версия программы, написанная на языке C. Компиляция ее исходного файла осуществлялась с передачей компилятору ключа, ответственного за генерацию максимально быстрого кода. Как видно на рис. 2, производительность GPU оказалось выше производительности CPU примерно в 16 раз.

Если ограничиваться расчетами с вещественными числами одинарной точности — *float*, то финальное ускорение достигнет уже 40 раз, так как GPU из-за конструктивных особенностей гораздо лучше оперирует с типом *float*, нежели с типом *double*. В отсутствие оптимизации преимущество GPU еще более велико: в расчетах с *double* ускорение составляет 26 раз, а в расчетах с *float* — 90 (см. рис. 3).

Ключи оптимизации заставляют компилятор генерировать код, в котором большая часть операций замкнута в регистровой и кэш памяти CPU, минуя медленную оперативную память. По этой причине в первом тесте отставание CPU не столь внушительно.

Сравнение эффективности различных программных сред почти всегда носит условный, а потому и дискуссионный характер. Оно зависит от множества факторов, начиная от версии операционной системы и модели оборудования, заканчивая решаемой задачей и качеством программного кода. В данном случае речь идет вообще о сравнении производительности совершенно разных по своей архитектуре вычислительных устройств. Понятно, что при использовании специализированных GPU или стандартов распараллеливания программ для многоядерных CPU, таких как OpenMP, результаты тестов поменяются в ту или иную сторону. Анализируя итоги проведенных тестов, мы лишь хотим сказать следующее: путем незначительной модификации последовательного кода, предназначенного для CPU, можно добиться очень существенного ускорения процесса численного интегрирования и решения других типичных задач при переносе расчетов на GPU, причем на GPU из той же ценовой категории, что и CPU. В этом смысле сравнение можно считать обоснованным.

По поводу переноса квадратурных формул Ньютона—Котеса на архитектуру GPU уместно сделать одно замечание. Допустим, в качестве метода интегрирования

выбран метод трапеций. В пределах элементарной ячейки $\Delta_{i,j}$ интеграл приближенно равен

$$S_{i,j} = \frac{h_x h_y}{4} (f_{i,j} + f_{i+1,j} + f_{i,j+1} + f_{i+1,j+1}). \quad (10)$$

Каждая нить будет вычислять выражение (10) и их суммировать. Узким местом такой организации вычислений является то, что функция f считается в любом внутреннем узле сетки четыре раза — по одному разу четырьмя смежными нитями. В условиях многомасштабных расчетов сказанное кратно снизит производительность GPU-программы, особенно когда в определении f находятся трансцендентные функции. Поэтому рекомендуется изначально исходить из составных формул с приведенными к общим множителям суммами

$$S = \sum_i \sum_j S_{i,j} = \dots + A \sum_i f_i + B \sum_i \sum_j f_{i,j} + \dots \quad (11)$$

Суммирование элементов векторов в (11) достаточно выполнять силами CPU или в крайних случаях путем редуцирования [20], а для суммирования элементов матриц можно воспользоваться описанным выше приемом.

4. Результаты численного моделирования. Применение вычислительных ресурсов GPU позволяет за приемлемое время и с удовлетворительной точностью находить интегральные члены в потенциале нековалентного взаимодействия атомов. В качестве примера при апробации методики расчетов были рассмотрены взаимодействия между атомами аргона, кремния и углерода. Полученные зависимости для электронных плотностей и межатомных потенциалов представлены на рис. 4, 5. Расчеты проводились для расстояний R из интервала 1–20 а.е. с шагом 0.05. Базовым методом при проведении численного интегрирования был выбран метод трапеций, как наиболее устойчивый к накоплению локальных ошибок. Точность расчетов при этом составляла 6 верных знаков после запятой и выдерживалась с помощью метода двойного подсчета.

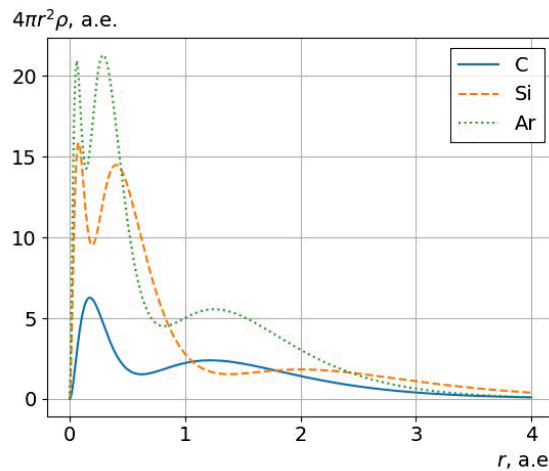


Рис. 4. Радиальные электронные плотности атомов

Интерполированием по расчетным данным удалось также определить параметры потенциальной ямы межатомного взаимодействия: R_0 — равновесное расстояние,

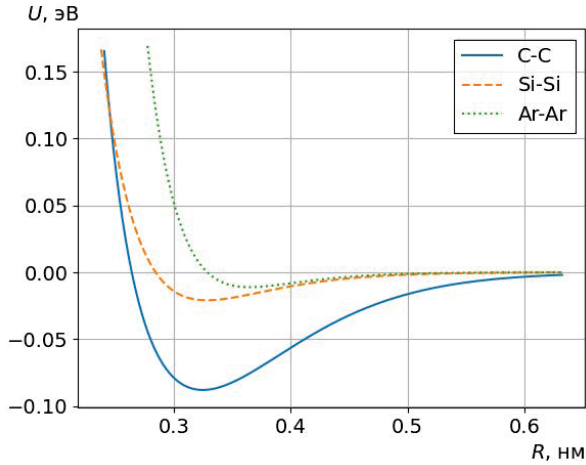


Рис. 5. Потенциалы межатомного взаимодействия

D — глубину ямы. Их значения указаны в таблице. Следует отметить, что сравнение с известными из литературы данными [14] для аргона дает хорошее совпадение: $R_0 = 0.363$ нм, $D = 0.011$ эВ.

Таблица. Параметры модельных потенциалов

Атомная пара	D , эВ	R_0 , нм	α , нм ⁻¹
Ar—Ar	0.011	0.365	20.104
C—C	0.021	0.329	16.162
Si—Si	0.088	0.325	11.244

Для атомов углерода и кремния полученные результаты удовлетворительно согласуются с расчетами константы Ван-дер-Ваальса C_6 [21]. Таким образом найденные параметры потенциальной ямы предлагается включать в модельные парные потенциалы типа Леннарда — Джонса

$$U(R) = D \left[\left(\frac{R_0}{R} \right)^{12} - 2 \left(\frac{R_0}{R} \right)^6 \right] \quad (12)$$

или Морзе

$$U(R) = D [\exp(2\alpha(R_0 - R)) - 2 \exp(\alpha(R_0 - R))], \quad (13)$$

где параметр α определяется по табулированным данным через жесткость межатомной связи $k_\alpha = U''(R_0) = 2D\alpha^2$. В дальнейшем модельные потенциалы (12) и (13) можно использовать в аналитических исследованиях, например при построении уравнений состояния вещества и расчете второго вириального коэффициента.

5. Заключение. В настоящей статье предложен и реализован новый алгоритм численного расчета парных межатомных потенциалов в теории электронного газа. Алгоритм включает в себя применение квадратурных формул, анализ подынтегральных функций электронных плотностей и процедуру распараллеливания вычислений на графических процессорах. Проведены вычисления нековалентного взаимодействия атомов аргона, кремния и углерода. На данных примерах продемонстрирована достаточно высокая эффективность алгоритма.

Предложенный подход к моделированию межатомного взаимодействия можно использовать для идентификации параметров модельных потенциалов Леннарда — Джонса и Морзе, что важно для многочисленных практических приложений.

Литература

1. Бахвалов Н. С., Жидков Н. П., Кобельков Г. М. Численные методы. М.: Наука, 1987. 600 с.
2. Калиткин Н. Н. Численные методы. М.: Наука, 1978. 512 с.
3. Жидков Е. П., Лобанов Ю. Ю., Рушай В. Д. Применение метода Ромберга для повышения точности вычисления кратных интегралов // Журнал вычислительной математики и математической физики. 2009. Т. 49. № 2. С. 232–240.
4. Рыбаков К. А. Точное вычисление погрешности аппроксимации кратных стохастических интегралов Ито // Сибирский журнал вычислительной математики. 2023. Т. 26. № 2. С. 205–213. <https://doi.org/10.15372/SJNM20230207>
5. Вьюсол Ю. А., Жорова А. Н., Муленко И. А., Хомкин А. Л. Применение методов компьютерной алгебры к расчету скобочных интегралов // Журнал вычислительной математики и математической физики. 2011. Т. 51. № 10. С. 1867–1882.
6. Galan-Garcia J. L., Rodriguez-Cielos P., Padilla-Dominguez Y., Galan-Garcia M. A., Atencia I., Rodriguez-Padilla P., Aguilera-Venegas G. SMIS: A stepwise multiple integration solver using a CAS // Mathematics. 2021. Vol. 9. N 22. P. 1–32. <https://doi.org/10.3390/math9222866>
7. Chernukha O., Bilushchak Y., Shakhovska N., Kulhanek R. A numerical method for computing double integrals with variable upper limits // Mathematics. 2022. Vol. 10. N 1. P. 1–26. <https://doi.org/10.3390/math10010108>
8. Горжуша О. А., Заводинский В. Г. О вычислении потенциала в многоатомных системах // Журнал вычислительной математики и математической физики. 2019. Т. 59. № 2. С. 325–333. <https://doi.org/10.1134/S0044466919020066>
9. Крутицкий П. А., Резниченко И. О. Улучшенная квадратурная формула для потенциала простого слоя // Журнал вычислительной математики и математической физики. 2023. Т. 63. № 2. С. 230–244. <https://doi.org/10.31857/S0044466923020114>
10. Соболев И. М. Многомерные квадратурные формулы и функции Хаара. М.: Наука, 1969. 288 с.
11. Раба Н. О. Разработка и реализация алгоритма расчета коагуляции в модели облаков со смешанной фазой с использованием технологии CUDA // Вестник Санкт-Петербургского университета. Сер. 10. Прикладная математика. Информатика. Процессы управления. 2011. Вып. 4. С. 94–104.
12. Fazanaro F. I., Soriano D. C., Suyama R., Madrid M. K., Oliveira J. R., Munoz I. B., Attux R. Numerical characterization of nonlinear dynamical systems using parallel computing: The role of GPUs approach // Communications in Nonlinear Science and Numerical Simulation. 2016. Vol. 37. P. 143–162. <https://doi.org/10.1016/j.cnsns.2015.12.021>
13. Ружников В. О. Повышение производительности расчета динамики частиц на параллельных системах // Вестник Санкт-Петербургского университета. Сер. 10. Прикладная математика. Информатика. Процессы управления. 2014. Вып. 1. С. 147–156.
14. Gordon R. G., Kim Y. S. Theory for the forces between closed-shell atoms and molecules // Journal of Chemical Physics. 1972. Vol. 56. N 6. P. 3122–3133. <https://doi.org/10.1063/1.1677649>
15. Waldman M., Gordon R. G. Scaled electron gas approximation for intermolecular forces // Journal of Chemical Physics. 1979. Vol. 71. N 3. P. 1325–1339. <https://doi.org/10.1063/1.438433>
16. Strand G., Bonham R. A. Analytical expressions for the Hartree–Fock potential of neutral atoms and for the corresponding scattering factors for X rays and electrons // Journal of Chemical Physics. 1964. Vol. 40. N 6. P. 1686–1691. <https://doi.org/10.1063/1.1725380>
17. Крылов В. И. Приближенное вычисление интегралов. М.: Наука, 1967. 500 с.
18. Рит М. Наноконструирование в науке и технике. Введение в мир нанорасчета. Ижевск: Регулярная и хаотическая динамика, 2005. 160 с.
19. Боресков А. В., Харламов А. А. Основы работы с технологией CUDA. М.: ДМК Пресс, 2010. 232 с.
20. Сандерс Дж., Кэндрот Э. Технология CUDA в примерах. Введение в программирование графических процессоров. М.: ДМК Пресс, 2011. 232 с.
21. Gould T., Bucko T. C6 coefficients and dipole polarizabilities for all atoms and many ions in rows 1–6 of the periodic table // Journal of Chemical Theory and Computation. 2016. Vol. 12. N 8. P. 3603–3613. <https://doi.org/10.1021/acs.jctc.6b00361>

Статья поступила в редакцию 27 апреля 2024 г.

Статья принята к печати 25 июня 2024 г.

Контактная информация:

Сокуров Аслан Артурович — канд. физ.-мат. наук; <https://orcid.org/0000-0001-9886-3602>,
isokuroff@mail.ru

Numerical calculation of integrals in the density functional theory in the electron gas approximation using CUDA technology

A. A. Sokurov

Kabardin-Balkar Scientific Center of the Russian Academy of Sciences, 89A, ul. Shortanova, Nalchik, 360000, Russian Federation

For citation: Sokurov A. A. Numerical calculation of integrals in the density functional theory in the electron gas approximation using CUDA technology. *Vestnik of Saint Petersburg University. Applied Mathematics. Computer Science. Control Processes*, 2024, vol. 20, iss. 3, pp. 335–349. <https://doi.org/10.21638/spbu10.2024.303> (In Russian)

The expression for the potential energy of interaction of two neutral atoms in the absence of a chemical bond consists of the sum of multiple and improper integrals. Due to the cumbersome nature of the functions, finding these integrals in explicit form is not possible. Software systems widely used in practice based on standard methods of computational mathematics are also not capable of providing satisfactory accuracy in their numerical calculations in a short time. In quantum chemistry and computational physics, the above greatly limits approaches to modeling the properties and structures of atomic/molecular systems. You have to rely on Monte-Carlo integration methods or formulas of the Gauss — Laguerre type, which are not so effective in terms of accuracy. In this article, in relation to this problem, we propose a method for transferring Newton — Cotes quadrature formulas to the architecture of graphic processors. The features of such a transfer are discussed in detail, designed to eliminate bottlenecks and maximize the performance of the corresponding calculation programs. The platform for massively parallel computing is CUDA technology from NVIDIA. Testing has shown that in typical tasks, the efficiency of programs for GPUs based on parallel analogues is on average an order of magnitude higher than classical ones. Within the framework of the proposed approach, it was possible to calculate interatomic interaction potentials in a wide range of changes in distances between atoms with high accuracy and in an acceptable computer time, as well as determine the equilibrium interaction parameters. The results obtained are in good agreement with the data known from the literature.

Keywords: parallel programming, numerical integration, CUDA technology, GPGPU, density functional theory.

References

1. Bahvalov N. S., Zhidkov N. P., Kobel'kov G. M. *Chislennye metody [Numerical methods]*. Moscow, Nauka Publ., 1987, 600 p. (In Russian)
2. Kalitkin N. N. *Chislennye metody [Numerical methods]*. Moscow, Nauka Publ., 1978, 512 p. (In Russian)
3. Zhidkov E. P., Lobanov Iu. Iu., Rushai V. D. Primenenie metoda Romberga dlia povysheniia tochnosti vychisleniia kratnykh integralov [Improving the accuracy of multiple integral evaluation by applying Romberg's method]. *Computational Mathematics and Mathematical Physics*, 2009, vol. 49, no. 2, pp. 232–240. (In Russian)
4. Rybakov K. A. Tochnoe vychislenie pogreshnosti approksimatsii kratnykh stokhasticheskikh integralov Ito [Exact calculation of the approximation error of multiple Ito stochastic integrals]. *Numerical Analysis and Applications*, 2023, vol. 26, no. 2, pp. 205–213. <https://doi.org/10.15372/SJNM20230207> (In Russian)
5. Vyzhol Iu. A., Zhorova A. N., Mulyenko I. A., Khomkin A. L. Primenenie metodov komp'yuternoi algebrы k raschetu skobochnykh integralov [Application of computer algebra for the calculation of bracket

integrals]. *Computational Mathematics and Mathematical Physics*, 2011, vol. 51, no. 10, pp. 1867–1882. (In Russian)

6. Galan-Garcia J. L., Rodriguez-Cielos P., Padilla-Dominguez Y., Galan-Garcia M. A., Atencia I., Rodriguez-Padilla P., Aguilera-Venegas G. SMIS: a stepwise multiple integration solver using a CAS. *Mathematics*, 2021, vol. 9, no. 22, pp. 1–32. <https://doi.org/10.3390/math9222866>

7. Chernukha O., Bilushchak Y., Shakhovska N., Kulhanek R. A numerical method for computing double integrals with variable upper limits. *Mathematics*, 2022, vol. 10, no. 1, pp. 1–26. <https://doi.org/10.3390/math10010108>

8. Gorkusha O. A., Zavodinskii V. G. O vychislenii potentsiala v mnogoatomnykh sistemakh [On the calculation of the interaction potential in multiatomic systems]. *Computational Mathematics and Mathematical Physics*, 2019, vol. 59, no. 2, pp. 325–333. <https://doi.org/10.1134/S0044466919020066> (In Russian)

9. Krutitskii P. A., Reznichenko I. O. Uluchshennaia kvadraturaia formula dlia potentsiala prostogo sloia [Improved quadrature formula for a single-layer potential]. *Computational Mathematics and Mathematical Physics*, 2023, vol. 63, no. 2, pp. 230–244. <https://doi.org/10.31857/S0044466923020114> (In Russian)

10. Sobol' I. M. *Mnogomernye kvadrurnye formuly i funktsii Khaara* [Multidimensional quadrature formulas and Haar functions]. Moscow, Nauka Publ., 1969, 288 p. (In Russian)

11. Raba N. O. Razrabotka i realizatsiia algoritma rascheta koaguliatsii v modeli oblakov so smeshanoi fazoi s ispol'zovaniem tekhnologii CUDA [Development and realization of algorithm of coagulation calculation in mixed-phase cloud model using CUDA technology]. *Vestnik of Saint Petersburg University. Series 10. Applied Mathematics. Computer Science. Control Processes*, 2011, iss. 4, pp. 94–104. (In Russian)

12. Fazanaro F. I., Soriano D. C., Suyama R., Madrid M. K., Oliveira J. R., Munoz I. B., Attux R. Numerical characterization of nonlinear dynamical systems using parallel computing: The role of GPU approach. *Communications in Nonlinear Science and Numerical Simulation*, 2016, vol. 37, pp. 143–162. <https://doi.org/10.1016/j.cnsns.2015.12.021>

13. Ruzhnikov V. O. Povyshenie proizvoditel'nosti rascheta dinamiki chastits na parallel'nykh sistemakh [Calculation performance improvement of particle dynamics on parallel systems]. *Vestnik of Saint Petersburg University. Series 10. Applied Mathematics. Computer Science. Control Processes*, 2014, iss. 1, pp. 147–156. (In Russian)

14. Gordon R. G., Kim Y. S. Theory for the forces between closed-shell atoms and molecules. *Journal of Chemical Physics*, 1972, vol. 56, no. 6, pp. 3122–3133. <https://doi.org/10.1063/1.1677649>

15. Waldman M., Gordon R. G. Scaled electron gas approximation for intermolecular forces. *Journal of Chemical Physics*, 1979, vol. 71, no. 3, pp. 1325–1339. <https://doi.org/10.1063/1.438433>

16. Strand G., Bonham R. A. Analytical expressions for the Hartree–Fock potential of neutral atoms and for the corresponding scattering factors for X rays and electrons. *Journal of Chemical Physics*, 1964, vol. 40, no. 6, pp. 1686–1691. <https://doi.org/10.1063/1.1725380>

17. Krylov V. I. *Priblizhennoe vychislenie integralov* [Approximate calculation of integrals]. Moscow, Nauka Publ., 1967, 500 p. (In Russian)

18. Rit M. *Nanokonstruirovaniie v nauke i tekhnike. Vvedenie v mir nanorascheta* [Nano-engineering in science and technology: an introduction to the world of nano-design]. Izhevsk, Reguliarnaia i khaoticheskaia dinamika Publ., 2005, 160 p. (In Russian)

19. Borekov A. V., Kharlamov A. A. *Osnovy raboty s tekhnologiei CUDA* [Basics of working with CUDA technology]. Moscow, DMK Press, 2010, 232 p. (In Russian)

20. Sanders Dzh., Kendrot E. *Tekhnologiiia CUDA v primerakh. Vvedenie v programmirovaniie graficheskikh protsessorov* [CUDA technology in examples. Introduction to GPU programming]. Moscow, DMK Press, 2011, 232 p. (In Russian)

21. Gould T., Bucko T. C6 coefficients and dipole polarizabilities for all atoms and many ions in rows 1–6 of the periodic table. *Journal of Chemical Theory and Computation*, 2016, vol. 12, no. 8, pp. 3603–3613. <https://doi.org/10.1021/acs.jctc.6b00361>

Received: April 27, 2024.

Accepted: June 25, 2024.

Author's information:

Aslan A. Sokurov — PhD in Physics and Mathematics; <https://orcid.org/0000-0001-9886-3602>, isokuroff@mail.ru