

Об использовании канонического разложения для визуализации результатов параметрических расчетов

А.К. Алексеев^{1,А,В}, А.Е. Бондарев^{2,А}, Ю.С. Пятакова^{3,В}

^А Институт прикладной математики им. М.В. Келдыша Российской академии наук
^В ПАО РКК Энергия им. С.П. Королева, Королев, Россия

¹ ORCID: 0000-0001-8317-8688, aleksey.k.alekseev@gmail.com

² ORCID: 0000-0003-3681-5212, bond@keldysh.ru

³ ORCID: 0000-0002-8055-7807, yuliya.pyatakova@rsce.ru

Аннотация

Аппроксимация тензора, возникающего при дискретизации многомерной функции, рассмотрена с точки зрения хранения и обработки результатов параметрических расчетов в задачах вычислительной аэрогазодинамики. Описан новый алгоритм расчета канонического разложения с помощью градиентного спуска для приближенно разложимого целевого функционала. Этот алгоритм использует случайный набор точек в гиперплоскости, ортогональной рассчитываемому ядру канонического разложения (“зонтик”), что позволяет его гибкое применение для аппроксимации тензоров с априорно неизвестным рангом и естественным образом переносится на такое тензорное разложение, как тензорный поезд. Представлены результаты численных экспериментов для модельных шестимерных функций и для ансамбля численных решений двумерных уравнений Эйлера. Эти уравнения описывают течение сжимаемого газа с двумя пересекающимися скачками уплотнения. В качестве параметров течения служат число Маха и углы отклонения потока. Результаты представлены для размерности 3 (просто численное решение) и 4 (набор численных решений в зависимости от числа Маха).

Ключевые слова: каноническое разложение, параметрические расчеты, вычислительная аэрогазодинамика, градиентная оптимизация, скользящая регуляризация.

Введение

Аппроксимация многомерных функций представляет интерес во многих приложениях, таких как квантовая механика, решение кинетических уравнений типа Больцмана, решение задач аэрогазодинамики в пространстве параметров.

В качестве примера приведем функцию $f(x, y, z, u, v, w)$ определенную в области $\Omega \subset R^6$ и соответствующую плотности распределения Больцмана. Для сетки, содержащей 100 узлов по каждой координате в шестимерном случае необходимо 10^{12} узлов памяти, содержащих числа. Для 64-значного числа это соответствует оперативной памяти 64 ТБ, что соответствует параметрам мощнейших современных суперкомпьютеров. Это значительно осложняет практические приложения, использующие уравнения Больцмана, как с точки стоимости расчетов, так и оперативности их применения.

В качестве другого примера рассмотрим набор численных решений задач аэрогазодинамики при аппроксимации в пространстве параметров. Это набор функций $f_i(x, y, z, \mathcal{Q}_1 \dots \mathcal{Q}_q)$, где $i = 1, \dots, p$ соответствует переменным течения (плотность, компоненты скорости, энергия), а $(\mathcal{Q}_1 \dots \mathcal{Q}_q) \subset \Omega_q \subset R^q$ соответствует параметрам задачи

(число Маха, Рейнольдса, угол атаки и т.д.). В распространенных практических приложениях эти функции распределены в пяти-семимерных пространствах, что подразумевает использование ресурсов, сравнимых с указанными в предыдущем примере.

С точки зрения практических приложений было бы крайне желательно данные задачи решать с вычислительными затратами на уровне ресурсов стандартного персонального компьютера.

Аппроксимация функций в многомерном пространстве представляет собой сложную задачу из-за экспоненциального роста требуемой оперативной памяти при увеличении размерности задачи (“проклятия размерности”). Одним из путей преодоления этих трудностей является использование тензорной формы многомерных задач и их аппроксимация с помощью тензорных разложений [1,2,3]. Этому вопросу и посвящена данная работа.

1. Каноническое разложение

Каноническое разложение $A = \sum_1^r Q_r^1 \otimes Q_r^2 \otimes \dots \otimes Q_r^d$ [1,2] является самым фундаментальным тензорным разложением (на его основе определяется ранг тензора). Здесь \otimes -внешнее (тензорное) произведение векторов, в индексной форме имеющее вид $A_{ij} = a_i \otimes b_j = a_i b_j$.

В индексном виде каноническое разложение записывается как

$$A_{i_1, i_2, \dots, i_d} = \sum_{\alpha=1}^r Q_{\alpha, i_1}^1 Q_{\alpha, i_2}^2 \dots Q_{\alpha, i_d}^d. \quad (1)$$

Это выражение единственно с точностью до перестановки членов и масштабирования [4,5].

Память, требуемая для хранения этой аппроксимации тензора $\sim rmd$ (где d - размерность пространства, n число узлов по одному направлению, r ранг тензора) (вместо полных затрат на хранение тензора n^d). Время для расчета одного узла $\sim rd$. Каноническое разложение обеспечивает гигантское сжатие данных (числа узлов сетки, используемых для аппроксимации функции в R^d), а именно $n^d \rightarrow d \cdot n \cdot r$.

Задача определения набора ядер $\{Q_{\alpha, i}^1, \dots, Q_{\alpha, k}^d\}$ в вариационной форме имеет вид

$$Q_{\beta}^n = \arg \min_{Q_{\beta}^n} \left\| A - \sum_1^r Q_{\alpha}^1 \otimes Q_{\alpha}^2 \otimes \dots \otimes Q_{\alpha}^d \right\|. \quad (2)$$

Ключевой величиной при использовании канонического разложения является ранг тензора r . По данным [1,4] она не вычислима из-за некорректности постановки задачи

$$r = \arg \min_r \left\| A - \sum_{\alpha=1}^r Q_{\alpha}^1 \otimes Q_{\alpha}^2 \otimes \dots \otimes Q_{\alpha}^d \right\|. \quad (3)$$

В соответствии с [1, 4] использование канонического разложения страдает от неустойчивостей и нуждается в регуляризации.

2. Методы расчета тензорных разложений

Методы расчета тензорных разложений можно с некоторой условностью разделить на два подкласса: методы, основанные на матризации тензора (линейной алгебре), например [3] и вариационные методы [5] (которые делятся на прямые и итерационные).

Методы, основанные на линейной алгебре, в существенной степени опираются на матризацию тензоров, сингулярное разложение и содержат много интересных и

оригинальных алгоритмов, позволяющих проводить операции прямо над ядрами, не обращаясь к аппроксимируемым функциям.

Вариационные формулировки, как правило, опираются на метод наименьших квадратов по переменным направлениям (alternating least squares (ALS)) [6,7], и, как правило, тоже используют матризацию тензоров. Как правило, для канонического разложения ALS реализуется с помощью произведения Хатри-Рао (Khatri Rao product, \odot) [6,7]). К сожалению, матризация тензора не приводит к его сжатию и не помогает против проклятия размерности.

Однако, в рамках вариационного подхода можно построить алгоритмы, которые не используют матризацию тензоров. Здесь мы рассмотрим оптимизационный метод, включающий в себя элементы метода наименьших квадратов по переменным направлениям и стохастический градиентный спуск. Его отличительными особенностями является использование специального множества (“зонтик”), на котором происходит оптимизация. Это множество позволяет разложить целевой функционал на сумму практически независимых функционалов. Другой особенностью является использование Тихоновской регуляризации нулевого порядка при определении ядер в скользящем по рангу варианте. При этой регуляризации компоненты ядер испытывают тем большее демпфирование, чем выше их ранг.

3. Градиентная оптимизация на случайном зонтике

Формально качество аппроксимации функции каноническим разложением на каждом шаге глобальной итерации может быть оценено с помощью следующей невязки (записанной для 6D)

$$\tilde{\varepsilon}_{total} = \sum_{i,j,k,l,m,p} \left\{ \sum_{\alpha=1}^r Q^x(\alpha,i) \cdot Q^y(\alpha,j) \cdot Q^z(\alpha,k) \cdot Q^u(\alpha,l) \cdot Q^v(\alpha,m) \cdot Q^w(\alpha,p) - \tilde{f}_{ijklmp} \right\}^2 / 2. \quad (4)$$

Здесь используются все узлы тензора, как в ALS с использованием произведения Хатри-Рао.

К сожалению, для рассматриваемых размерностей этот функционал напрямую невычислим (по крайней мере на обычных персональных компьютерах) из-за огромных затрат компьютерного времени. Мы его значение численно оценивали с помощью метода Монте-Карло в форме

$$\varepsilon_{MC} = \frac{1}{2Mc} \sum_{s=1}^{s=MC} \left\{ \sum_{\alpha} Q^x(\alpha,i) \cdot Q^y(\alpha,j) \cdot Q^z(\alpha,k) \cdot Q^u(\alpha,l) \cdot Q^v(\alpha,m) \cdot Q^w(\alpha,p) - \tilde{f}_{ijklmp} \right\}^2, \quad (5)$$

где на каждом шаге суммирования s каждый индекс из i, j, k, l, m, p выбирался случайным и равномерно распределенным числом. В результате мы вычисляли среднюю по ансамблю сумму квадратов ошибки аппроксимации. Число попыток в ансамбле находилось в интервале $MC = 1000 \div 100000$, где результат довольно слабо менялся при изменении числа попыток.

Так как функционал (4) практически не вычислим, а функционал (5) может быть применен к расчету ядер только в рамках метода Монте-Карло, необходима альтернатива. В этом качестве здесь мы используем специальную форму целевого функционала, не требующую больших вычислительных усилий и позволяющую получать выражения для градиента целевого функционала по компонентам ядер, пригодные для градиентной оптимизации. В отличие от стандартного подхода, основанного на произведении Хатри-Рао, предлагаемый алгоритм не ограничен размерностью при использовании (не использует матризацию тензора или полный набор узлов тензора) и существенно проще алгоритмически.

Основой этого алгоритма является специальный выбор ансамбля точек, на которых производится оптимизация. Эти точки случайным образом выбираются на гиперплоскости, ортогональной координате, соответствующей ядру (на “зонтике”, Рис. 1)).

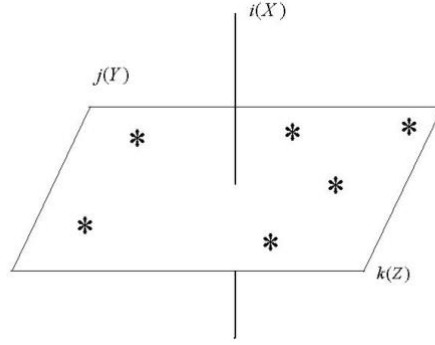


Рис. 1. Ансамбль точек, на котором производится оптимизация (“зонтик”)

Ядра $Q^x(\alpha, i), Q^y(\alpha, j), Q^z(\alpha, k)...$ получаютcя последовательным перебором.

Рассмотрим алгоритм на примере ядра разложения по x в точке $i: Q^x(\alpha, i), \alpha = 1...r$ и соответствующую невязку $\varepsilon_{x,i}(Q^x(*, i))$, (остальные элементы ядра $Q^x(\alpha, i)$ определяются последовательно по i)

$$\varepsilon_{x,i}(Q^x) = 1/2 \sum_{e=1...Lens} \left\{ \left(\sum_{\alpha} C(\alpha) \cdot Q^x(\alpha, i) \cdot Q^y(\alpha, j_e^i) \cdot Q^z(\alpha, k_e^i) \dots - \tilde{f}_{ij_e^i k_e^i l_e^i m_e^i p_e^i} \right) \cdot \left(\sum_{\beta} C(\beta) \cdot Q^x(\beta, i) \cdot Q^y(\beta, j_e^i) \cdot Q^z(\beta, k_e^i) \dots - \tilde{f}_{ij_e^i k_e^i l_e^i m_e^i p_e^i} \right) \right\}. \quad (6)$$

Точки ансамбля $j_e^i, k_e^i, l_e^i, m_e^i, p_e^i$ нумеруем индексом $e = 1...Lens$ (i соответствует узлу раскрытия “зонтика”). Они расположены в гиперплоскости, ортогональной x (в точке i) и выбираются случайным образом.

Здесь $\tilde{f}_{ij_e^i k_e^i l_e^i m_e^i p_e^i}$ точное значение функции в точке $i, j_e^i, k_e^i, l_e^i, m_e^i, p_e^i$.

В расчетах размер ансамбля выбирается много большим предварительной оценки ранга тензора $Lens \gg r$. Увеличение размера ансамбля увеличивает вычислительные затраты на расчет невязки и градиента.

Возмутим это ядро на величину $\Delta Q^x(\beta, i)$. Соответствующее возмущение невязки имеет вид:

$$\Delta \varepsilon_{x,i}(Q^x) = \sum_{e=1...Lens} \left\{ \left(\sum_{\alpha} C(\alpha) \cdot Q^x(\alpha, i) \cdot Q^y(\alpha, j_e^i) \cdot Q^z(\alpha, k_e^i) \dots - \tilde{f}_{ij_e^i k_e^i l_e^i m_e^i p_e^i} \right) \cdot \left(\sum_{\beta} C(\beta) \cdot \Delta Q^x(\beta, i) \cdot Q^y(\beta, j_e^i) \cdot Q^z(\beta, k_e^i) \dots - \tilde{f}_{ij_e^i k_e^i l_e^i m_e^i p_e^i} \right) \right\}. \quad (7)$$

Соответственно градиент невязки имеет вид:

$$\partial \varepsilon_{x,i} / \partial Q^x(\beta, i) = \sum_{e=1...Lens} \left\{ \left(\sum_{\alpha} C(\alpha) \cdot Q^x(\alpha, i) \cdot Q^y(\alpha, j_e^i) \cdot Q^z(\alpha, k_e^i) \dots - \tilde{f}_{ij_e^i k_e^i l_e^i m_e^i p_e^i} \right) \cdot C(\beta) \cdot Q^y(\beta, j_e^i) \cdot Q^z(\beta, k_e^i) \dots \right\}. \quad (8)$$

Для перекрестных производных (по другим ядрам, здесь для примера, по $Q^y(\beta, j)$) в точках ансамбля e соответствующее выражение имеет вид:

$$\partial \varepsilon_{x,i} / \partial Q^y(\beta, j) = \left\{ \left(\sum_{\alpha} C(\alpha) \cdot Q^x(\alpha, i) \cdot Q^y(\alpha, j_e^i) \cdot Q^z(\alpha, k_e^i) \dots - \tilde{f}_{ij_e^i k_e^i l_e^i m_e^i p_e^i} \right) \cdot C(\beta) \cdot Q^x(\beta, i) \cdot Q^z(\beta, k_e^i) \dots \right\}. \quad (9)$$

при $j = j_e^i$ (иначе- 0). Важно, что суммирование по ансамблю пропадает. Поэтому, такие члены примерно в $1/Lens$ раз меньше основного (8). На этом основании мы ими пренебрегаем, что позволяет проводить оптимизацию только по $Q^x(\beta, i)$.

Таким образом, вместо одной глобальной невязки (используемой в расчетах для контроля)

$$\varepsilon_{\Sigma} = \left\{ \sum_i \varepsilon_{x,i} + \sum_j \varepsilon_{y,j} + \dots \right\} / (6N \cdot L_{ens}) \quad (10)$$

мы оптимизируем $6N$ отдельных невязок (считаем, что по каждой координате у нас N узлов)

$$\varepsilon_{x,i}; \varepsilon_{y,j}; \dots \quad (11)$$

Компоненты ядер ищутся последовательной оптимизацией невязок по номеру ядра и номеру узла на ядре

$$Q^c(\beta, i) = \arg \min \varepsilon_{c,i}(Q^c(\beta, i)) \quad (12)$$

Для оптимизации мы используем градиентный спуск, для $Q^x(\beta, i)$ имеющий вид:

$$\{Q^x(\beta, i)\}^{n+1} = \{Q^x(\beta, i)\}^n - \tau \nabla_{x,i,\beta} \varepsilon. \quad (13)$$

Общая структура алгоритма имеет вид (цикл по ядрам и их узлам):

<p>Алгоритм 1. Поиск ядер канонического разложения с помощью наискорейшего спуска</p> <p>Input Initialize for $c = 1 \dots d$ (X,Y,Z,U,V,W) do ! цикл по координатам for $i = 1 \dots N_c$ do ! цикл по узлам вдоль ядра Поиск минимума реализуется методом наискорейшего спуска for $\beta = 1 \dots r$ do ! цикл по индексу ранга $(Q^c(\beta, i))^{n+1} = (Q^c(\beta, i))^n - \tau_{c,i} \cdot \partial \varepsilon_{c,i} / \partial Q^c(\beta, i)$ end for output</p>
--

Такую примитивную структуру алгоритма можно реализовать из-за того, что градиенты приближенно разложимы:

$$\nabla \varepsilon_{\Sigma} = \{ \nabla_{Q_x^i} \varepsilon_{x,i} + \nabla_{Q_y^j} \varepsilon_{y,j} + \dots + \nabla_{Q_x^i} \varepsilon_{x,i} + \nabla_{Q_y^j} \varepsilon_{y,j} + \dots \} + O(1/L_{ens}). \quad (14)$$

с точностью до $\sim O(1/L_{ens})$. Причина разложимости- выбор ансамбля точек, на которых производится расчет функционала невязки.

3.1 Скользящая регуляризация

В работе использована скользящая по индексу ранга регуляризация, использующая Тихоновскую регуляризацию нулевого порядка [9], с коэффициентом регуляризации, увеличивающимся по индексу ранга β в виде

$$\varepsilon_{c,i} + \gamma Q^c(\beta, i)^2 \cdot (\beta - 1)^2 / 2 \quad (15)$$

Здесь γ - базовый коэффициент регуляризации.

Соответствующие итерации имеют вид:

$$(Q^c(\beta, i))^{n+1} = (Q^c(\beta, i))^n - \tau_{c,i} \cdot \{ \partial \varepsilon_{c,i} / \partial Q^c(\beta, i) + \gamma \cdot Q^c(\beta, i) \cdot (\beta - 1)^2 \} \quad (16)$$

Общая оптимизация сводится к локальной последовательной оптимизации по $\sim 6N$ невязкам. Для контроля сходимости отслеживаются величины ε_{MC} (5) и ε_{Σ} (10).

Оптимизация элементов ядер заканчивалась при $\varepsilon \leq \varepsilon_1$, $\varepsilon_1 = 10^{-7} \div 10^{-9}$. Процесс построения канонического разложения заканчивался при $\varepsilon_{MC} \leq \varepsilon_2$, $\varepsilon_2 = 10^{-5} \div 10^{-7}$.

4. Результаты численных экспериментов по интерполяции многомерных функций

Рассмотрим аппроксимацию шестимерной функции \tilde{f} (точнее говоря тензора \tilde{f}_{ijklmp} , соответствующего значениям функции в узлах регулярной сетки) с помощью канонического разложения и соответствующего набора ядер $Q^x(\alpha, i) \cdot Q^y(\alpha, j) \cdot Q^z(\alpha, k) \cdot Q^u(\alpha, l) \cdot Q^v(\alpha, m) \cdot Q^w(\alpha, p)$.

В численных экспериментах использовалась сетка, содержащая 100 узлов по каждой координате. С формальной точки зрения при такой сетке хранение \tilde{f}_{ijklmp} потребует 10^{12} ячеек памяти, что совершенно нереалистично ни с точки зрения хранения, ни с точки зрения расчетов. В качестве иллюстрации, отметим, что в этом случае память, занимаемая ядрами с рангом 100 (достаточно большим) составляет $6 \cdot 100 \cdot 100 = 60000$ ячеек, что иллюстрирует сверхвысокую степень сжатия информации ($\sim 10^7$) при использовании канонического разложения.

В процессе отладки проводилось сравнение численных (полученных прямым численным дифференцированием) и аналитических градиентов (полученных с помощью выражения (8)) показавшее их практически полное совпадение.

Результаты расчетов дают достаточно устойчивые и воспроизводимые оценки.

На первом этапе проведено тестирование аппроксимации различных функций с помощью канонического разложения. Рассмотрены следующие многомерные функции, расположенные в порядке возрастания сложности.

$$f = x + y \quad (17)$$

Это предельно простая функция, но ее априорный ранг не известен. Использовано 30 итераций, ранг вариировался от одного до 10. Зависимость величины невязки (5) от величины ранга представлена в Табл. 1.

Таблица 1. Зависимость невязки (5) от ранга для функции (17).

ранг	1	2	3	4	5	10
невязка	$1.46 \cdot 10^{-2}$	$7.13 \cdot 10^{-8}$	$1.68 \cdot 10^{-7}$	$2.22 \cdot 10^{-7}$	$4.39 \cdot 10^{-7}$	$2.00 \cdot 10^{-5}$

Характерно быстрое падение невязки при достижении истинного значения ранга (здесь- 2), и медленный рост при дальнейшем увеличении ранга. Из этих результатов видно, что, в целом, величина невязки может служить индикатором истинного ранга тензора. По мере увеличения ранга увеличивается зашумленность результата, по всей видимости, это отражает неустойчивости, возникающие вследствие некорректности для канонического разложения [1,4].

На Рис. 2 и 3 представлена точная функция (17) и ее аппроксимация при ранге 2.

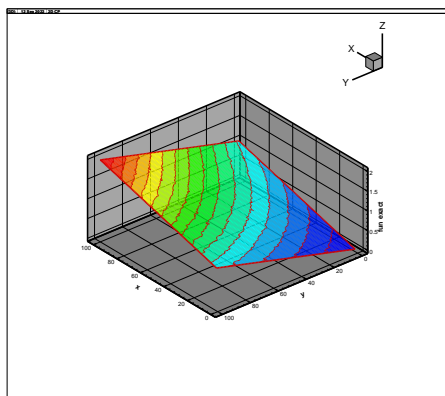


Рис. 2. Точная функция

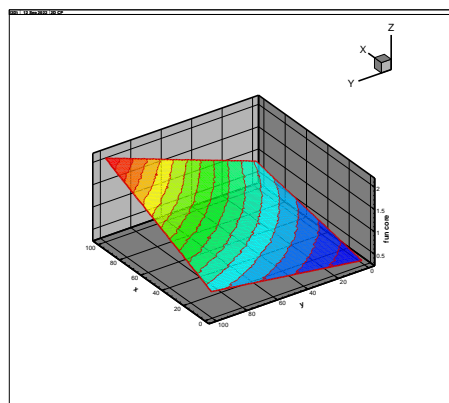


Рис. 3. Аппроксимация с помощью канонического разложения

Функция (17) является двумерной, рассмотрим более сложный истинно шестимерный случай, задаваемый гауссианой и суммой синусов по всем направлениям в виде

$$f = 5 \cdot \exp(-rad^2) + \sin(x/5) + \sin(y/5) + \sin(z/5) + \sin(u/5) + \sin(v/5) + \sin(w/5) \quad (18)$$

$$(rad = 0.001 \cdot ((ix - 50) + (iy - 50) + (iz - 50) + (iu - 50) + (iv - 50) + (iw - 50))).$$

На рис. 4 и 5 представлены результаты для ранга 10 и ансамбля из 1000 точек в плоскости x, y , остальные переменные соответствуют серединам интервалов на сетке 100. Результаты аппроксимации практически неотличимы от точной функции. При уменьшении размеров ансамбля качество аппроксимации уменьшается. В целом при расчетах достаточно сложных функций ансамбль должен быть существенно больше ожидаемого ранга. К счастью использование Тихоновской регуляризации нулевого порядка в скользящем варианте позволяет минимизировать текущий ранг и избавляет от необходимости его априорного знания (или перебора рангов).

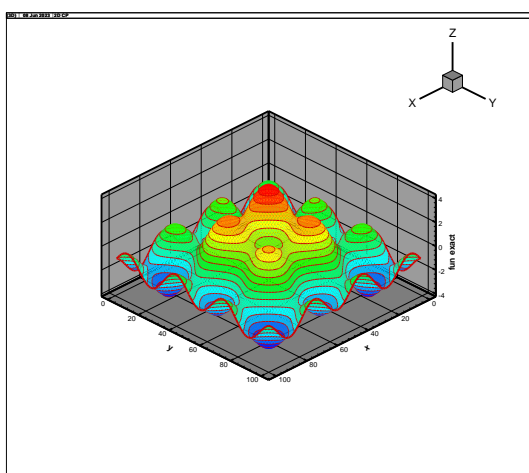


Рис. 4. Точная функция (18)

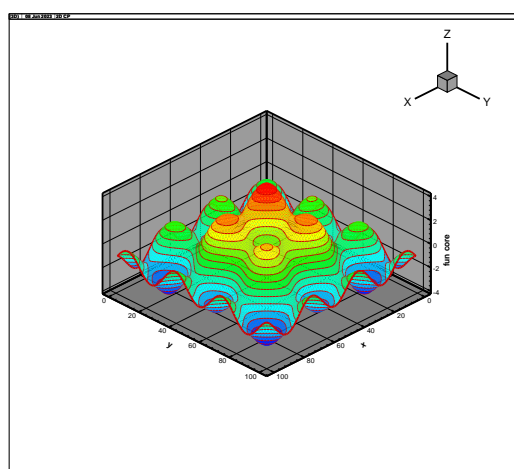


Рис. 5. Аппроксимация (18) с помощью канонического разложения

На Рис. 6 представлено поведение разных критериев сходимости в зависимости от числа итераций. В логарифмической форме представлены невязки ε_{Σ} (10) (Eps), глобальная невязка, оцененная с помощью Монте-Карло ε_{MC} (5) (Eps_MC) и норма градиента невязки (grad norm).

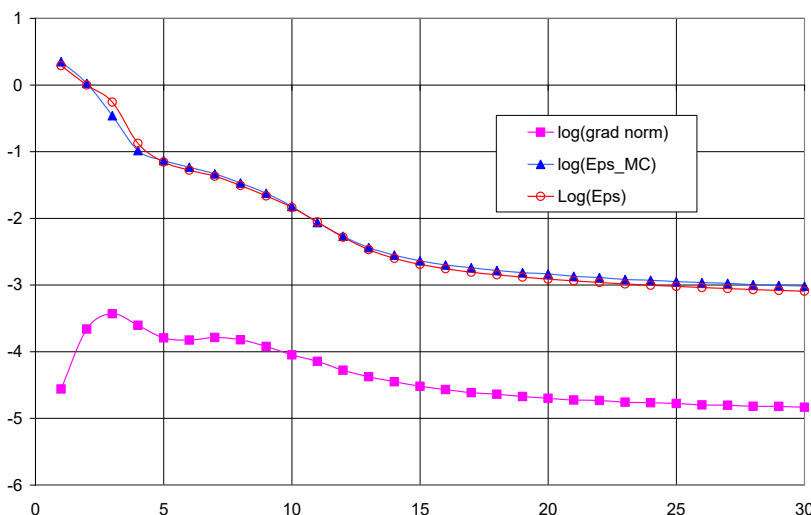


Рис. 6. Разные критерии сходимости в зависимости от числа итераций.

Невязки ε_Σ (10) (Eps) и глобальная невязка, оцененная с помощью Монте-Карло ε_{MC} (5) (Eps_MC), демонстрируют близкое поведение, пригодное для контроля сходимости. Норма градиента невязки (grad norm) на начальных итерациях ведет себя немонотонно, что ограничивает ее применимость в качестве критерия останова.

Строго говоря, используемый метод не предусматривает монотонной минимизации глобальных критериев качества, так как минимизация осуществляется поточно. Однако, численные эксперименты демонстрируют монотонную сходимость глобальных критериев. Данный алгоритм обеспечивает заметно более быструю сходимость в сравнении с алгоритмом, использованным в [10].

В реальности в численных экспериментах мы получаем не истинный ранг функции (тензора), а аппроксимацию ранга r_ε , обеспечивающую невязку ε . Рис. 7 демонстрирует, что используемый в работе вариант Тихоновской регуляризации [9] позволяет уменьшить эффективный ранг за счет увеличения параметра регуляризации, но при этом качественно меняет ядра.

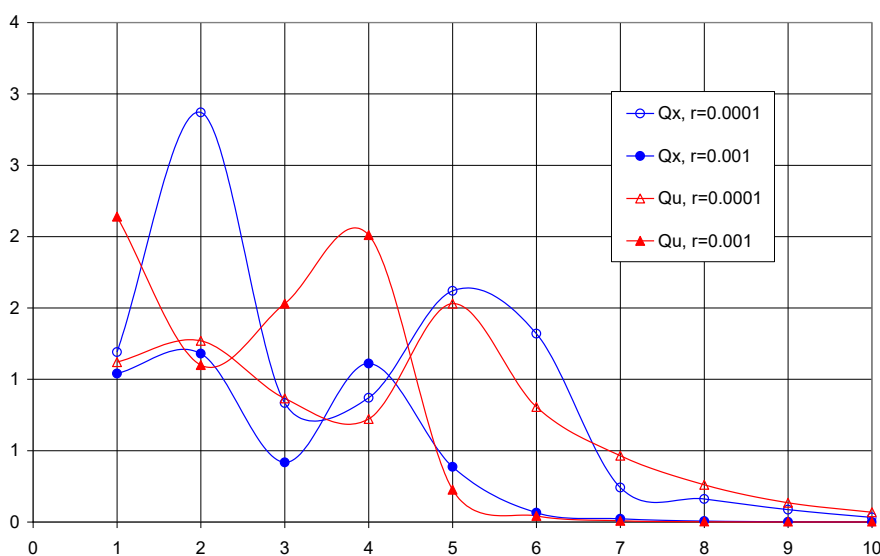


Рис. 7. Зависимость невязки от величины ранга для функции (18)

5. Результаты численных экспериментов по аппроксимации результатов параметрических расчетов

В качестве примера рассмотрим двумерное стационарное поле течения в пространстве параметров размерностью от одного до трех. Оно соответствует случаю взаимодействия двух плоскопараллельных сверхзвуковых потоков воздуха, образующихся при обтекании симметричных клиньев. Структура течения описывается стационарными Маховским и регулярным режимами взаимодействия ударных волн. Параметрами является число Маха $M \in (2, 6)$ и углы отклонения потока $(\theta_1, \theta_2) \in (15, 30)$.

Набор расчетов на одной пространственной сетке с разными параметрами (θ_1, θ_2, M) можно рассматривать как тензор порядка 6: $T_{pi,j;k,n,m} = (\rho_{i,j;k,n,m}, u_{i,j;k,n,m}, v_{i,j;k,n,m}, e_{i,j;k,n,m})$ (p - номер газодинамической переменной, i, j индексы узлов по координатам, k, n, m - номера параметров (θ_1, θ_2, M)).

Рассмотрим построение канонического разложения на таком ансамбле решений, позволяющего радикальное сжатие данных.

Используя структуру данных типа “зонтик” будем строить каноническое разложение. Один наш расчет поля течения (элементарный “строительный блок”) $T_{p;i,j;1,1,1}$ - это полностью заполненный тензор порядка 3.

Можно достаточно легко построить тензор порядка четыре используя последовательность тензоров порядка три (переходя с $T_{p;i,j;1,1,1}$ на $T_{p;i,j;2,1,1}$ и так далее). Это достаточно трудоемко ($I_{ens} = 10 \div 100$ отдельных расчетов), но реализуемо. Однако уже при построении тензора порядка пять, возникнет проблема – на каждом слое надо заново набирать тензор ранга 4 снова, что вычислительно крайне трудоемко, тем более это не реализуемо для тензора следующего ранга. Таким образом, возникает дополнительная задача-восстановление всего тензора по неполным данным, которая также может рассматриваться в описанной выше оптимизационной постановке со специальным выбором опорного ансамбля расчетов.

На Рис. 8 и 9 представлены результаты расчета поля течения (плотности) для $\theta_1 = 15^\circ$ (отклонение верхнего потока), $\theta_2 = 20^\circ$ (отклонение нижнего потока), $M = 4$ и соответствующей аппроксимации с помощью канонического разложения (ранг 15, число точек ансамбля-2000). При этом использован тензор порядка три $T_{p;i,j}$. Направление течения справа, на переднем плане – зона повышенной плотности за пересекающимися скачками уплотнения.

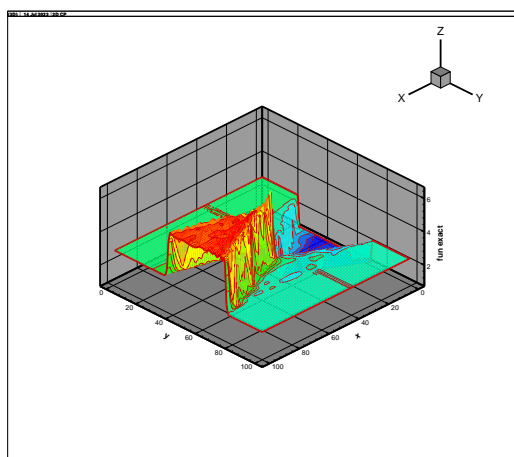


Рис. 8. Расчет поля плотности

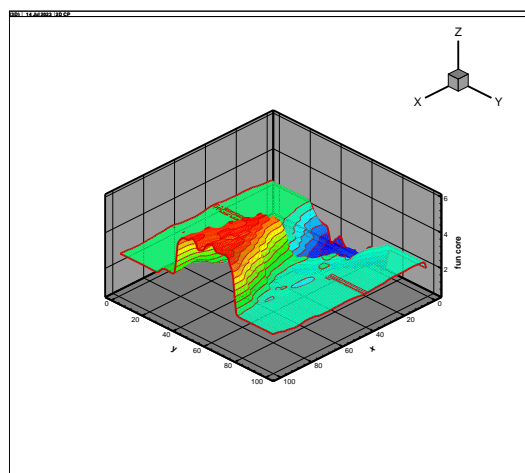


Рис. 9. Аппроксимация поля плотности с помощью канонического разложения (тензор порядка три)

Результаты, представленные на Рис. 8 и 9 демонстрируют работоспособность аппроксимации (сжатия) полей течения с помощью канонического разложения.

На Рис. 10 и 11 представлены результаты расчета поля течения (плотности) ($\theta_1 = 15^\circ$, $\theta_2 = 20^\circ$, $M = 3$) и соответствующей аппроксимации с помощью канонического разложения (ранг 3, число точек ансамбля-200). При этом использован тензор порядка четыре: $T_{p;i,j,M}$, в котором использовано пять слоев по $M = (3.0; 3.5; 4.0; 4.5; 5.0)$.

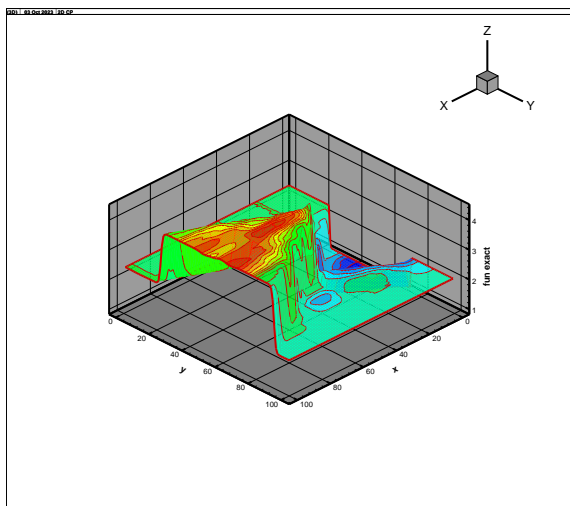


Рис. 10. Расчет поля плотности

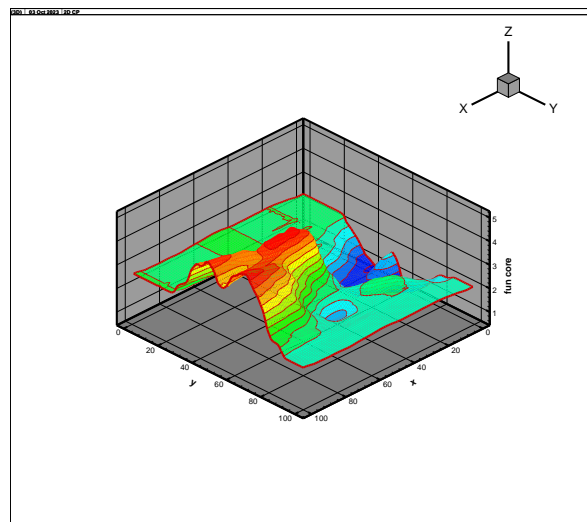


Рис. 11. Аппроксимация поля плотности с помощью канонического разложения (тензор порядка четыре)

Результаты, представленные на Рис. 10 и 11 демонстрируют работоспособность аппроксимации (сжатия) однопараметрического поля течения с помощью канонического разложения. Наблюдаемые осцилляции аппроксимации вероятно связаны с некорректностью задачи определения канонического разложения. Есть определенная надежда, что при переходе к тензорному поезду качество решения улучшится.

6. Обсуждение

В связи с некорректностью задачи [4] при определении ядер канонического разложения предусмотрена возможность квадратичной регуляризации нулевого порядка по Тихонову [9] со скользящей регуляризацией, позволяющей сжимать аппроксимацию ранга r_ε , обеспечивающую невязку ε (получать оценку ранга сверху).

Тензорные разложения достаточно активно используются для целей визуализации [11], поскольку позволяют построить легко вычислимую модель, аппроксимирующую труднообрабатываемый набор данных в параметрическом пространстве. Тензорные разложения (каноническое разложение, тензорный поезд, иерархический Таккер) используются для экономичного решения многомерных задач типа уравнения Больцмана [5,13,14]. Часто для этих целей используется формат тензорного поезда и кросс-аппроксимация [13]. Рассматриваемый здесь подход легко расширяется на формат тензорного поезда и лишен ограничений на вычислительные ресурсы, характерных для кросс-аппроксимации (связанных с матризацией тензора). Используемое здесь каноническое разложение позволяет эффективно аппроксимировать и хранить многомерные функции. Затраты компьютерного времени на работу с функциями в шестимерном пространстве (при использовании 100 узлов по каждой координате, что формально требует хранения и работу с 10^{12} чисел) составляют 1-3 минуты на ПК (процессор Intel I5, 2.66 ГГц) при затратах памяти на хранение ядер в максимальном случае около 10^5 чисел. Есть надежда, что переход к формату тензорного поезда (планируемый в дальнейших работах) позволит преодолеть часть недостатков канонического разложения, связанных с некорректностью и неустойчивостью.

Заключение

Предложен оптимизационный алгоритм определения ядер в каноническом разложении, требования по используемой памяти которого, существенно меньше, чем в стандартных методах, использующих матризацию тензора и произведение Хатри-Рао. В алгоритме используется скользящая по рангу регуляризация, которая позволяет понижать эффективный ранг аппроксимации за счет снижения норм слоев высоких индексов ранга.

Численные эксперименты показывают, что применение данного алгоритма для реализации канонического разложения позволяет хранить и визуализировать функции в многомерном пространстве с очень умеренными затратами памяти и времени счета. Это представляет особый интерес с точки зрения работы с результатами многопараметрических расчетов задач аэрогазодинамики.

Библиографический список

1. W. Hackbusch. *Tensor Spaces and Numerical Tensor Calculus*. Springer, 2012.
2. H. Yorick, S. Willi-Hans, *Matrix Calculus, Kronecker Product and Tensor Product: A Practical Approach to Linear Algebra, Multilinear Algebra and Tensor Calculus with Software I*, WOS 2019
3. Oseledets I. V., Tensor-train decomposition, *SIAM J. Sci. Comput.*, 33 (2011), pp. 2295–2317
4. V. D. Silva and L.-H. Lim, Tensor rank and the ill-posedness of the best low rank approximation problem, *SIAM J. Matrix Anal. Appl.*, 30(3) 1084–1127, 2008.
5. A. M. P. Boelens, D. Venturi, D. M. Tartakovsky, Parallel tensor methods for high-dimensional linear PDEs, *J. Computat. Phys.* 375 (2018) 519–539.
6. Shuangzhe Liu, Gaotz Trenkler, Hadamard, Khatri-Rao, Kronecker and other matrix products, *Int. J. Information and System Sciences*, 2008, Volume 4, Number 1, Pages 160–177
7. P. Comon, X. Luciani, and A.L.F. De Almeida, Tensor decompositions, alternating least squares and other tales, *Journal of Chemometrics*, vol. 23, no. 7–8, pp. 393–405, 2009.
8. A. Uschmajew, Local convergence of the alternating least squares algorithm for canonical tensor approximation, *SIAM J. Matrix Anal. Appl.* 33 (2) (2012) 639–652,
9. S. Rabanser, O. Shchur, S. Gunnemann, *Introduction to Tensor Decompositions and their Applications in Machine Learning*, arXiv:1711.10781v1, 2017
9. Тихонов А.Н., Арсенин В.Я., *Методы решения некорректных задач*, М., Наука, 1979.
10. Алексеев А.К., Бондарев А.Е., Пятакова Ю.С., О визуализации функций в многомерном пространстве с помощью канонического разложения, *Научная визуализация*, т. 14, № 3, с. 73–91, 2022
11. Yannis Panagakis, Jean Kossaifi, Grigorios G. Chrysos, James Oldfield, Mihalis A. Nicolaou, Anima Anandkumar and Stefanos Zafeiriou, *Tensor Methods in Computer Vision and Deep Learning*, arXiv:2107.03436v1, 2021
12. Oseledets I., Tyrtyshnikov E., TT-cross approximation for multidimensional arrays, *Linear Algebra Appl.*, 432 (2010), pp. 70–88.
13. Arnout M. P. Boelens, Daniele Venturi, Daniel M. Tartakovsky, Tensor methods for the Boltzmann-BGK equation, arXiv:1911.04904v2 2020
14. A.V. Chikitkin, E.K. Kornev, V.A. Titarev, Numerical solution of the Boltzmann equation with S-model collision integral using tensor decompositions, arXiv:1912.04582v1 2019

On Application of Canonical Decomposition for the Visualization of Results of Multiparameter Computations

A.K. Alekseev^{1,A,B}, A.E. Bondarev^{2,A}, Yu. S. Pyatakova^{3,B}

^A Keldysh Institute of Applied Mathematics RAS

^B RSC Energia, Korolev, Russia

¹ ORCID: 0000-0001-8317-8688, aleksey.k.alekseev@gmail.com

² ORCID: 0000-0003-3681-5212, bond@keldysh.ru

³ ORCID: 0000-0002-8055-7807, yuliya.pyatakova@rsce.ru

Abstract

The approximation of the tensor appearing at a discretization of the multidimensional function is considered from the viewpoint of storing and treating of the results of parametric computations obtained in computational aerogas dynamics. The new algorithm for the computation of the canonical decomposition using gradient descent and approximately decomposable goal functional is described.

This algorithm applies the random set of points on the hyperplane orthogonal to the computed core of the canonical decomposition (“umbrella”) that ensures its flexible application for an approximation of the tensors with a priori unknown rank and may be naturally transferred on such tensor decomposition as the tensor train. The results of the numerical tests are presented for the model six-dimensional functions and for an ensemble of the numerical solutions for the two-dimensional Euler equations. These equations describe the flow of the compressible gas with two crossing shock waves. The Mach number and angles of the flow deflection serve as the flow parameters. The results are provided for the dimensionality 3 (simple numerical solution) and 4 (the ensemble of the numerical solutions in dependence on the Mach number).

Keywords: canonical decomposition, parametric calculations, computational aerogas dynamics, gradient optimization, sliding regularization.

References

15. W. Hackbusch. Tensor Spaces and Numerical Tensor Calculus. Springer, 2012.
16. H. Yorick, S. Willi-Hans, Matrix Calculus, Kronecker Product and Tensor Product: A Practical Approach to Linear Algebra, Multilinear Algebra and Tensor Calculus with Software I, WOS 2019
17. Oseledets I. V., Tensor- train decomposition, SIAM J. Sci. Comput., 33 (2011), pp. 2295–2317
18. V. D. Silva and L.-H. Lim, Tensor rank and the ill-posedness of the best low rank approximation problem, SIAM J. Matrix Anal. Appl., 30(3) 1084–1127, 2008.
19. A. M. P. Boelens, D. Venturi, D. M. Tartakovsky, Parallel tensor methods for high-dimensional linear PDEs, J. Computat. Phys. 375 (2018) 519–539.
20. Shuangzhe Liu, Gaotz Trenkler, Hadamard, Khatri-Rao, Kronecker and other matrix products, Int. J. Information and system Sciences, 2008, Volume 4, Number 1, Pages 160–177
21. P. Comon, X. Luciani, and A.L.F. De Almeida, Tensor decompositions, alternating least squares and other tales, Journal of Chemometrics, vol. 23, no. 7–8, pp. 393–405, 2009.
- A. Uschmajew, Local convergence of the alternating least squares algorithm for canonical tensor approximation, SIAM J. Matrix Anal. Appl. 33 (2) (2012) 639–652,
22. S. Rabanser, O. Shchur, S. Gunnemann, Introduction to Tensor Decompositions and their Applications in Machine Learning, arXiv:1711.10781v1, 2017

23. Tikhonov, A.N., Arsenin, V.Y.: Solutions of Ill-Posed Problems. Winston and Sons, Washington DC (1977).
24. A.K. Alekseev, A.E. Bondarev, Yu.S. Pyatakova, On the visualization of multidimensional functions using canonical decomposition, Scientific Visualization, 2022, volume 14, number 3, pages 73 - 91
25. Yannis Panagakis, Jean Kossaifi, Grigorios G. Chrysos, James Oldfield, Mihalis A. Nicolaou, Anima Anandkumar and Stefanos Zafeiriou, Tensor Methods in Computer Vision and Deep Learning, arXiv:2107.03436v1, 2021
26. Oseledets I., Tyrtshnikov E., TT-cross approximation for multidimensional arrays, Linear Algebra Appl., 432 (2010), pp. 70–88.
27. Arnout M. P. Boelens, Daniele Venturi, Daniel M. Tartakovsky, Tensor methods for the Boltzmann-BGK equation, arXiv:1911.04904v2 2020
28. A.V. Chikitkin, E.K. Kornev, V.A. Titarev, Numerical solution of the Boltzmann equation with S-model collision integral using tensor decompositions, arXiv:1912.04582v1 2019