

# Быстрое численное решение краевых задач с известной функцией Грина через циклическую свертку

Т. В. АБРАМОВ

Институт нефтегазовой геологии и геофизики им. А.А. Трофимука СО РАН, Новосибирск, Россия

Новосибирский государственный университет, Россия

Контактный e-mail: AbramovTV@ipgg.sbras.ru

Предложен алгоритм быстрого численного решения краевых задач, для которых известно явное выражение функции Грина. Их решение в этом случае можно найти как свертку этой функции с правой частью уравнения. Прямой алгоритм вычисления свертки требует  $O(N^2)$  операций. В работе предложен способ понижения вычислительной сложности до  $O(N \log N)$ , для чего расчетная функция приводится к виду циклической свертки, которая может быть вычислена быстро с помощью различных алгоритмов.

Работа по оптимизации вычислений производилась на примере решения задачи моделирования неустойчивости Рэлея — Тейлора в высоковязкой ньютоновской жидкости. Для соответствующей краевой задачи в полупространстве со свободной поверхностью известно аналитическое выражение функции Грина. Предложенный алгоритм позволил существенно (на порядок и более) ускорить вычисления и может быть применен для других задач.

*Ключевые слова:* циклическая свертка, теорема о свертке, быстрое преобразование Фурье, краевая задача, функция Грина, неустойчивость Рэлея — Тейлора.

## Введение

В вычислительной математике очень важное значение имеют алгоритмы быстрого вычисления сверток (БВС). Они позволяют кардинально уменьшить сложность многих вычислительных операций вида “каждое-с-каждым” от  $O(N^2)$  до  $O(N \log N)$ . Не претендуя на полноту ссылок, укажем лишь быстрый алгоритм умножения полиномов и чисел [1], фильтрацию сигналов (изображений) [2], моделирование гравитационного взаимодействия  $N$ -тел [3].

Большинство алгоритмов БВС ориентированы только на циклическую свертку [4]. Однако некоторые другие операции, например линейную свертку, можно вычислять через циклическую, и это дает преимущества в скорости [5]. Приведение какой-либо операции к виду циклической свертки — нетривиальная задача, которая ограничивает возможности использования алгоритмов БВС в конкретном приложении.

Численное решение краевой задачи, для которой известна функция Грина, также представляет собой сумму, похожую на циклическую свертку, но не эквивалентную ей.

Автору удалось привести вычисления к формальной циклической свертке путем некоторых преобразований входных и выходных данных. Полученная методика апробирована на реальной и практически значимой физической задаче моделирования неустойчивости Рэлея—Тейлора в высоковязкой ньютоновской жидкости, где было достигнуто значительное (на порядок и более) повышение скорости расчетов.

## 1. Постановка задачи

Для обыкновенных дифференциальных уравнений второго порядка рассмотрим краевую задачу с однородными краевыми условиями

$$\begin{aligned} y'' + p_1(x)y' + p_2(x)y &= r(x), \quad a \leq x \leq b, \\ y(a) &= y(b) = 0, \end{aligned} \quad (1)$$

где  $y(x)$  — искомая функция, а  $p_1(x)$ ,  $p_2(x)$  — известны на отрезке  $[a, b]$ . Решение (1) можно получить в виде

$$y(x) = \int_a^b G(x, s)r(s)ds, \quad (2)$$

где  $G : [a, b] \times [a, b] \rightarrow \mathbb{R}$  — функция Грина краевой задачи (1) с однородными граничными условиями;  $x \in [a, b]$ ,  $s \in [a, b]$  [6].

Рассмотрим только дискретную постановку задачи. Положим  $x_n = a + n\Delta x$ ,  $s_m = a + m\Delta x$ ,  $ds = \Delta x$ ,  $n, m = 0, \dots, N - 1$ ,  $\Delta x(N - 1) = b - a$ . В этом случае формула (2) принимает вид

$$y(a + n\Delta x) = \Delta x \sum_{m=0}^{N-1} G(a + n\Delta x, a + m\Delta x)r(a + m\Delta x), \quad n = 0, \dots, N - 1. \quad (3)$$

Для краткости можно переписать это выражение следующим образом:

$$y(n) = \Delta x \sum_{m=0}^{N-1} G(n, m)r(m), \quad n = 0, \dots, N - 1, \quad (4)$$

считая, что  $y(n)$ ,  $G(n, m)$ ,  $f(m)$  — дискретные функции. Очевидно, что непосредственное (прямое)<sup>1</sup> вычисление  $y(n)$  по формуле (4) имеет сложность  $O(N^2)$ . Требуется минимизировать количество этих операций.

## 2. Метод решения

Рассмотрим две ограниченные дискретные периодические функции  $f : \mathbb{Z} \rightarrow \mathbb{R}$  и  $g : \mathbb{Z} \rightarrow \mathbb{R}$  с периодом  $N$ , т. е.  $f(n) = f(kN + n)$  и  $g(n) = g(kN + n)$ ,  $k \in \mathbb{Z}$ . Дискретной циклической сверткой  $f$  и  $g$  называется сумма вида

$$y(n) = f(n) * g(n) = \frac{1}{N} \sum_{m=0}^{N-1} f(n - m)g(m), \quad n = 0, \dots, N - 1. \quad (5)$$

<sup>1</sup>Здесь и в дальнейшем термины “непосредственный” и “прямой” будут применяться только к алгоритмам вычисления сумм вида (4) и аналогичным, а не к решению краевых задач в целом.

Из-за цикличности функций  $f$  и  $g$  уравнение (5) будет эквивалентно

$$y(n) = f(n) * g(n) = \frac{1}{N} \sum_{m=0}^{N-1} f((N+n-m)_{\text{mod } N})g(m), \quad n = 0, \dots, N-1, \quad (6)$$

где  $X_{\text{mod } Y}$  означает остаток от деления  $X$  на  $Y$ . В формуле (6) формально можно считать, что  $f(n)$  и  $g(n)$  — непериодические функции, определенные на интервале  $[0, N-1]$ .

Дискретная циклическая свертка (6) примечательна тем, что существует множество алгоритмов ее вычисления за время, меньшее чем  $O(N^2)$  [4]. На этом основан предложенный метод быстрого расчета выражений вида (4). Он заключается в том, чтобы доопределить  $G$  и  $f$  до некоторых периодических  $\hat{G}$  и  $\hat{f}$  так, что их циклическая свертка дает искомую сумму (4).

Выбор наилучшего алгоритма для БВС не является главной целью данной работы, предполагается только, что он имеет сложность, меньшую чем  $O(N^2)$ . Для решения практической задачи, описанной в разд. 5, реализован “классический” алгоритм, основанный на теореме о свертке [4, 5]. Его сложность составляет  $O(N \log N)$ , как и многих других алгоритмов БВС.

### 3. Приведение к циклической свертке

Основное отличие суммы вида (4) от (6) состоит в том, что  $G(n, m)$  — функция двух переменных. Допустим, что она выражается только через  $n - m$ , т. е.

$$y(n) = \Delta x \sum_{m=0}^{N-1} G(n-m)f(m), \quad n = 0, \dots, N-1. \quad (7)$$

При этом область определения  $G(n)$  с  $[0, N-1] \times [0, N-1]$  поменяется на  $[-(N-1), (N-1)]$ , т. е., чтобы вычислить  $y(n)$  в  $N$  отсчетах, нужно задать  $G(n)$  в  $2N-1$  отсчетах. Выражение (7) еще не циклическая свертка, потому что  $G$  в общем случае не является периодической функцией с периодом  $N$ . Тем не менее  $f(n)$  и  $G(n)$  можно формально преобразовать в некоторые функции  $\hat{f}(n)$  и  $\hat{G}(n)$  так, что их циклическая свертка даст нужный результат. Определим  $\hat{G}(n)$  как дискретную функцию на  $2N-1 = N^*$  отсчетах

$$\hat{G}(n) = \begin{cases} G(n), & 0 \leq n < N, \\ G(n-N^*), & N \leq n < N^*, \end{cases} \quad (8)$$

и  $\hat{f}(n)$  как дискретную функцию также из  $N^*$  отсчетов

$$\hat{f}(n) = \begin{cases} f(n), & 0 \leq n < N, \\ 0, & N \leq n < N^*. \end{cases} \quad (9)$$

В дальнейшем потребуется также  $G(n)$  выразить через  $\hat{G}(n)$ :

$$G(n) = \begin{cases} \hat{G}(n), & 0 \leq n < N, \\ \hat{G}(N^*+n), & -(N-1) \leq n < 0. \end{cases} \quad (10)$$

Выполнив циклическую свертку  $\hat{G}(n)$  и  $\hat{f}(n)$ , получим

$$\hat{y}(n) = \hat{G}(n) * \hat{f}(n) = \frac{1}{N^*} \sum_{m=0}^{N^*-1} \hat{G}((N^* + n - m)_{\text{mod } N^*}) \hat{f}(m), \quad n = 0 \dots N^* - 1. \quad (11)$$

Поскольку  $\hat{f}(m) = 0$  при  $n \geq N$ , верхний предел суммы (11) можно заменить на  $N - 1$ :

$$\hat{y}(n) = \hat{G}(n) * \hat{f}(n) = \frac{1}{N^*} \sum_{m=0}^{N-1} \hat{G}((N^* + n - m)_{\text{mod } N^*}) \hat{f}(m), \quad n = 0 \dots N^* - 1. \quad (12)$$

Рассмотрим результат выполнения свертки на половине области определения, а точнее, при  $n < N$ . Очевидно, что  $|n - m| < N^*$  и возможны два случая:

$$(N^* + n - m)_{\text{mod } N^*} = \begin{cases} n - m, & n - m \geq 0, \\ N^* + n - m, & n - m < 0. \end{cases} \quad (13)$$

Перепишем (12) в соответствии с (13) при  $n < N$ :

$$\hat{y}(n) = \hat{G}(n) * \hat{f}(n) = \begin{cases} \frac{1}{N^*} \sum_{m=0}^{N-1} \hat{G}(N^* + n - m) \hat{f}(m), & n - m < 0, \\ \frac{1}{N^*} \sum_{m=0}^{N-1} \hat{G}(n - m) \hat{f}(m), & n - m \geq 0, \end{cases} \quad n = 0, \dots, N - 1. \quad (14)$$

Функцию  $\hat{G}(n)$  в обоих случаях можно заменить на  $G(n)$  согласно (10):

$$\hat{y}(n) = \hat{G}(n) * \hat{f}(n) = \begin{cases} \frac{1}{N^*} \sum_{m=0}^{N-1} G(n - m) \hat{f}(m), & n - m < 0, \\ \frac{1}{N^*} \sum_{m=0}^{N-1} G(n - m) \hat{f}(m), & n - m \geq 0, \end{cases} \quad n = 0, \dots, N - 1. \quad (15)$$

При  $m < N$  функция  $\hat{f}(m)$  также эквивалентна  $f(m)$  в соответствии с (9). Поэтому сумма (15) окончательно примет вид

$$\hat{y}(n) = \hat{G}(n) * \hat{f}(n) = \frac{1}{N^*} \sum_{m=0}^{N-1} G(n - m) f(m), \quad n = 0, \dots, N - 1. \quad (16)$$

Выражение (16) справедливо для  $n < N$  и эквивалентно (7) с точностью до постоянного множителя. Другими словами, выполнив формально свертку  $\hat{f}(n)$  и  $\hat{G}(n)$ , получим искомую функцию в “первой половине” ( $n < N$ ) области определения  $\hat{y}(n)$ .

#### 4. Трехмерная постановка

Формулы (8) и (9) можно обобщить для пространств различной мерности. Приведем без вывода выражения  $\hat{f}(n_1, n_2, n_3)$  и  $\hat{G}(n_1, n_2, n_3)$  для трехмерного пространства:

$$\hat{G} = \begin{cases} G(n_1, n_2, n_3), & n_1 < d_1, n_2 < d_2, n_3 < d_3; \\ G(n_1 - 2d_1 + 1, n_2, n_3), & n_1 \geq d_1, n_2 < d_2, n_3 < d_3; \\ G(n_1, n_2 - 2d_2 + 1, n_3), & n_1 < d_1, n_2 \geq d_2, n_3 < d_3; \\ G(n_1, n_2, n_3 - 2d_3 + 1), & n_1 < d_1, n_2 < d_2, n_3 \geq d_3; \\ G(n_1 - 2d_1 + 1, n_2 - 2d_2 + 1, n_3), & n_1 \geq d_1, n_2 \geq d_2, n_3 < d_3; \\ G(n_1 - 2d_1 + 1, n_2, n_3 - 2d_3 + 1), & n_1 \geq d_1, n_2 < d_2, n_3 \geq d_3; \\ G(n_1, n_2 - 2d_2 + 1, n_3 - 2d_3 + 1), & n_1 < d_1, n_2 \geq d_2, n_3 \geq d_3; \\ G(n_1 - 2d_1 + 1, n_2 - 2d_2 + 1, n_3 - 2d_3 + 1), & n_1 \geq d_1, n_2 \geq d_2, n_3 \geq d_3, \end{cases} \quad (17)$$

где  $G(n_1, n_2, n_3)$  задана на трехмерной регулярной сетке  $d_1 \times d_2 \times d_3$ . Функция  $\hat{G}(n_1, n_2, n_3)$  будет задана на сетке  $(2d_1 - 1) \times (2d_2 - 1) \times (2d_3 - 1)$  узлов. Формула для  $\hat{f}(n_1, n_2, n_3)$  выражается проще:

$$\hat{f} = \begin{cases} f(n_1, n_2, n_3), & n_1 < d_1, n_2 < d_2, n_3 < d_3, \\ 0, & \text{иначе.} \end{cases} \quad (18)$$

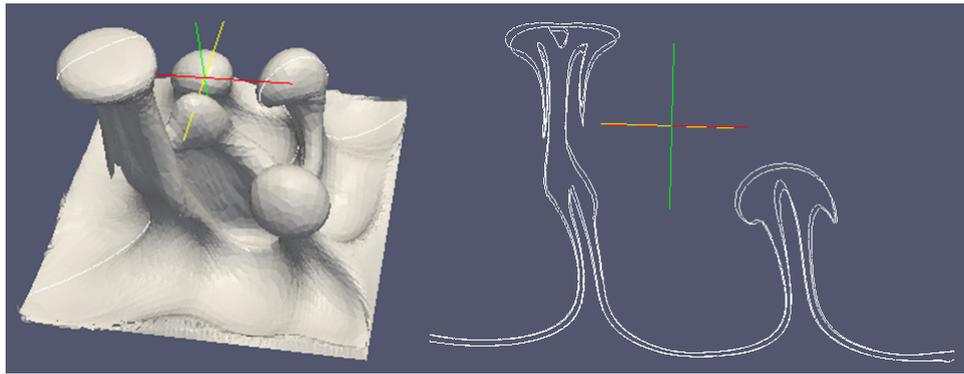
Функция  $\hat{f}(n_1, n_2, n_3)$  также задана на сетке  $(2d_1 - 1) \times (2d_2 - 1) \times (2d_3 - 1)$  узлов. Размер сетки вырастает в  $2^k$  раз для  $k$ -мерного пространства, но благодаря невысокой алгоритмической сложности метода вычисления производятся значительно быстрее, особенно на больших сетках. Для пространств других размерностей формулы можно вывести по аналогии.

#### 5. Результаты моделирования

Предложенный метод апробирован на решении конкретной физической задачи. Требовалось ускорить моделирование неустойчивости Рэля — Тейлора в высоковязкой несжимаемой несмешивающейся ньютоновской жидкости. Для соответствующей краевой задачи известно аналитическое выражение функции Грина [7]. Уже был реализован и отлажен прямой алгоритм вычисления (4) в 2D постановке [8] и в 3D с использованием технологии параллельных вычислений CUDA на графических ускорителях [9]. Однако квадратичная зависимость времени от размерности задачи не давала значительно увеличивать густоту расчетных сеток, даже при расчете на гибридном вычислительном кластере.

Применение описанного метода значительно ускорило решение задачи. Для вычисления свертки был применен “классический” алгоритм, использующий теорему о свертке [4]. Быстрые дискретные преобразования Фурье, необходимые для работы алгоритма, рассчитывались на графическом ускорителе NVIDIA Tesla M2090 с помощью библиотеки cuFFT.

Для верификации с предыдущими версиями программного обеспечения, использующими прямой алгоритм для (4), рассчитаны эволюции нескольких моделей двумя методами на одном ускорителе Tesla M2090. На рисунке представлена конечная стадия



Конечная стадия эволюции модели: слева — поверхность гравитационно-неустойчивого слоя легкой жидкости, всплывающего вверх через более тяжелые, с образованием грибообразных структур; справа — наложены два разреза неустойчивого слоя, один получен при расчете прямым методом, а другой предложенным (быстрым). Расчетная сетка  $100 \times 100 \times 100$  узлов

эволюции одной из моделей, точнее, только поверхность гравитационно-неустойчивого слоя легкой жидкости, всплывающего вверх через более тяжелые, образуя сложные грибообразные структуры. Эта модель была рассчитана (“выращена”) из одного и того же исходного состояния двумя различными методами, поэтому на рисунке справа показаны в наложении два идентичных разреза модели — для предложенного (быстрого) и прямого методов.

В табл. 1 показаны времена расчетов полной эволюции моделей для двух методов на различных сетках.

Прямой алгоритм можно считать точным, поскольку он позволяет считать скорость течения в любой точке независимо, в данном случае только на границах несмешивающихся жидкостей. Быстрый метод, основанный на теореме о свертке, дает поле течения только на регулярной сетке, поэтому требуется интерполяция для вычисления скорости на границах (в данном случае линейная). Из-за этого на разрезах (см. рисунок) заметно некоторое расхождение результатов. Однако при увеличении густоты сетки до  $150 \times 150 \times 150$  узлов разрезы были визуальными неотличимы.

Из табл. 1 очевидно преимущество быстрого метода по скорости. Важно также и то, что для прямого метода время расчета зависит не квадратично от размера сетки,

Т а б л и ц а 1. Время расчета эволюции моделей на одном ускорителе Tesla M2090, мин

Размер сетки	Прямой метод	Предложенный метод	Ускорение (разы)
$75 \times 75 \times 75$	11.4	1.05	10.86
$100 \times 100 \times 100$	27.0	2.84	9.51
$150 \times 150 \times 150$	144	12.5	11.52

Т а б л и ц а 2. Время расчета поля течения по регулярной сетке на одном ускорителе Tesla M2090, с

Размер сетки	Прямой метод	Предложенный метод	Ускорение (разы)
$64 \times 64 \times 64$	18	1.49	12.08
$128 \times 128 \times 128$	1133	2.61	434.10
$256 \times 256 \times 128$	18157	29.04	625.24

так как расчет скорости производится только на границах поверхностей. Это не только увеличивает точность, но и уменьшает порядок сложности. Иногда все же бывает предпочтительней вычислять поле течения или напряженно-деформированное состояние по регулярной сетке. В этом случае быстрый метод будет намного эффективней. В табл. 2 показано, сколько времени занимает расчет одного шага эволюции по регулярной сетке.

## Заключение

Предложен метод для быстрого численного решения краевых задач с известной функцией Грина. Он особенно эффективен, когда решение требуется получить на регулярной сетке с большим количеством узлов ( $10^6$  и более). Важно, что описанный способ приведения суммы (4) к циклической свертке, а также большинство алгоритмов быстрого вычисления циклической свертки являются точными, а не приближенными. Таким образом, если пренебречь ошибками округления в численных расчетах, результат будет таким же, как при непосредственном суммировании по формуле (4).

С помощью предложенного метода удалось существенно ускорить решение важной для практики физической задачи, относящейся к нефтегазовой геологии [8, 9]. На сетках порядка  $10^6$  узлов ускорение составило один порядок, но благодаря тому, что алгоритмическая сложность кардинально снижается с квадратичной до линейно-логарифмической, преимущество будет более значительным на более густых сетках.

Подводя итог, можно сказать, что преимуществами метода являются:

- высокая скорость счета, определяемая выбранным алгоритмом БВС;
- возможность использования различных библиотек для БВС;
- точность решения зависит только от ошибок округления при вычислении свертки.

К особенностям метода, ограничивающим сферу его применения, можно отнести:

- возможность расчета только по регулярной сетке;
- увеличение размеров расчетной сетки в  $2^k$  раз, где  $k$  — размерность пространства.

Первый недостаток не критичен, если использовать хорошую интерполяцию, а в некоторых случаях не является недостатком вовсе. Второй более значителен, поскольку достаточно большая задача может не поместиться целиком в оперативной памяти компьютера, от этого скорость работы значительно упадет. На практике эти недостатки не мешали решению задачи, они окупались высокой эффективностью алгоритма.

**Благодарности.** Работа выполнена в рамках Программы VIII.73.2 фундаментальных научных исследований СО РАН.

## Список литературы / References

- [1] **Кнут Д.Э.** Искусство программирования. Т. 2. Получисленные алгоритмы: Учеб. пособие. 3-е изд.; Пер. с англ. М.: Изд. дом “Вильямс”, 2000. 832 с.  
**Knuth, D.E.** The art of computer programming. Vol. 2. Seminumerical algorithms. 3rd edition. New Jersey: Pearson Education (US), 1997. 784 p.
- [2] **Гонсалес Р., Вудс Р.** Цифровая обработка изображений: Пер. с англ. М.: Техносфера, 2005. 1072 с.  
**Gonzalez, R.C., Woods, R.E.** Digital image processing. 2nd edition. Prentice Hall, 2002. 793 p.

- [3] Numerical methods in astrophysics: An introduction / P. Bodenheimer, G.P. Laughlin, M. Rozyczka, H.W. Yorke, T. Plewa. London: Taylor & Francis Ltd, 2006. 344 p.
- [4] **Нуссбаумер Г.** Быстрое преобразование Фурье и алгоритмы вычисления свертки: Пер. с англ. М.: Радио и связь, 1985. 247 с.  
**Nussbaumer, H.J.** Fast Fourier transform and convolution algorithms: Second corrected and updated edition. Berlin: Springer-Verlag, 1982. 276 p.
- [5] Линейная и циклическая свертка. Адрес доступа: <http://www.dsplib.ru/content/conv/conv.html> (дата обращения 21.07.2015).  
The linear and cyclic convolution. Available at: <http://www.dsplib.ru/content/conv/conv.html> (accessed 21.07.2015). (In Russ.)
- [6] **Эльсгольд Л.Э.** Дифференциальные уравнения и вариационное исчисление. М.: Наука, 1969. 424 с.  
**Elsgolts, L.** Differential equations and the calculus of variations. Miami: University Press of the Pacific, 2003. 444 p.
- [7] **Лунёв Б.В.** Изостазия как динамическое равновесие вязкой жидкости // Докл. АН СССР. 1986. Т. 290, № 1. С. 72–76.  
**Lunev, B.V.** Isostasy as the dynamic equilibrium of a viscous fluid // Doklady AN SSSR. 1986. Vol. 290, No. 1. P. 72–76. (In Russ.)
- [8] **Лунёв Б.В., Лапковский В.В.** Быстрое численное моделирование соляной тектоники: возможность оперативного использования в геологической практике // Физическая мезомеханика. 2009. Т. 12, № 1. С. 63–74.  
**Lunev, B.V., Lapkovskii, V.V.** Fast numerical simulation of salt tectonics: possibility of the operational application in geological practice // Physical Mesomechanics. 2009. Vol. 12, No. 1. P. 63–74. (In Russ.)
- [9] **Абрамов Т.В.** Массивно-параллельный расчет неустойчивости Рэлея—Тейлора с помощью аналитического выражения функции Грина соответствующей краевой задачи // Вычисл. технологии. 2015. Т. 20, № 4. С. 3–16.  
**Abramov, T.V.** Massively parallel Rayleigh—Taylor instability simulation using analytical expression of Green’s function of the corresponding boundary value problem // Comput. Technologies. 2015. Vol. 20, No. 4. P. 3–16. (In Russ.)

*Поступила в редакцию 5 ноября 2015 г.*

### **Fast numerical solution of boundary value problems with known Green’s function using cyclic convolution**

ABRAMOV, TIMOFEY V.

Trofimuk Institute of Petroleum-Gas Geology and Geophysics SB RAS, Novosibirsk, 630090, Russia

Novosibirsk State University, Novosibirsk, 630090, Russia

Corresponding author: Abramov, Timofey V., e-mail: [AbramovTV@ipgg.sbras.ru](mailto:AbramovTV@ipgg.sbras.ru)

Author has developed a method for fast numerical solution of boundary value problems with known explicit form of Green’s function. Solution of this kind of problems can be established just as an integral of product of two known functions, rather than

solution of a difference problem. In the discrete formulation, we have just some kind of sum to calculate, but naive algorithm requires  $O(N^2)$  operations, where  $N$  is the number of nodes of the computational grid. This procedure is extremely inefficient when  $N$  is large enough, and parallel computing doesn't significantly help when  $N$  is about  $10^6$  and greater.

The performance of the procedure could be improved substantially due to the modification of the algorithm. The desired sum (solution of the problem) has been converted to the form of cyclic convolution, which can be calculated in  $O(N \log N)$  operations. The proposed method relies on this conversion only, and it does not imply any specific algorithm for calculating a cyclic convolution. In practice, the convolution theorem is often used for this. It uses both forward and inverse fast discrete Fourier transforms.

The method has been numerically implemented for the problem of simulating Rayleigh — Taylor instability in high viscous incompressible and immiscible Newtonian fluid in the three-dimensional domain. It delivers the increase of the calculation speed by several orders of magnitude compared to the naive summation. The developed program uses convolution theorem for cyclic convolution. Fourier transforms have been calculated using NVIDIA's graphics accelerators with the help of cuFFT library.

*Keywords:* cyclic convolution, convolution theorem, fast Fourier transform, boundary value problem, Green's function, Rayleigh — Taylor instability.

**Acknowledgements.** The work was conducted as part of the Program of Basic Scientific Research VIII.73.2 SB RAS.

*Received 5 November 2015*