

ПРИМЕНЕНИЕ ДРОБНО-РАЦИОНАЛЬНЫХ
ПРИБЛИЖЕНИЙ И МЕТОДА КОЛЛОКАЦИИ ДЛЯ
РЕШЕНИЯ УРАВНЕНИЯ БЮРГЕРСА

И.А. БУГОЕЦ, Б.В. СЕМИСАЛОВ, В.П. ШАПЕЕВ

Представлено

Abstract: An algorithm is proposed for solving initial-boundary value problems for Burgers equation. The algorithm is based on the use of rational barycentric interpolations, the collocation method, and a new scheme for time approximation of the solution that employs the relaxation method. An adaptive approach for taking into account the location of singular point of analytic continuation of the solution is developed. Solutions of a test problem on the interval $[-1, 1]$ with the initial condition $-\sin \pi x$ are obtained for viscosity values down to $\nu = 10^{-8}$.

Keywords: Burgers equation, rational barycentric interpolation, collocation method, relaxation method, adaptive algorithm, singular point, low viscosity.

BUGOETS, I.A., SEMISALOV, B.V., SHAPEEV, V.P., APPLICATION OF RATIONAL APPROXIMATIONS AND COLLOCATION METHOD FOR SOLVING THE BURGERS EQUATION.

© 2026 БУГОЕЦ И.А., СЕМИСАЛОВ Б.В., ШАПЕЕВ В.П..

Работа И.А. Бугоецца выполнена при поддержке Математического Центра в Академгородке, соглашение с Министерством науки и высшего образования Российской Федерации (№ 075-15-2025-348), работа Б.В. Семисалова выполнена в рамках государственного задания Института математики им. С.Л. Соболева СО РАН (проект № FWNF-2026-0028). Авторы выражают благодарность Брындину Луке Сергеевичу за предоставленный пакет программ и обсуждение реализации метода.

Поступила 1 января 2023 г., опубликована 31 декабря 2023 г.

1 Описание проблемы и подходов к её решению

Численный анализ нелинейных дифференциальных уравнений, описывающих сложные гидродинамические процессы представляет существенный интерес как с точки зрения фундаментальной науки, так и для множества приложений. В этом контексте универсальным примером, используемым для проверки точности, устойчивости и скорости сходимости численных методов, является уравнение Бюргерса (УБ) [1]¹

$$u_t + uu_x = \nu u_{xx}. \quad (1)$$

Здесь ν – кинематическая вязкость, $u(t, x)$ – скорость течения жидкости, t – время (далее полагаем $t \in [0, T]$), x – координата частицы жидкости.

Особый интерес к УБ со стороны специалистов в области гидродинамики связан с тем, что это уравнение является достаточно простым и содержит две ключевые компоненты, характеризующие нелинейную динамику вязкой жидкости, – конвективный и вязкий члены, т. е. компоненты uu_x и νu_{xx} . В связи с этим анализ УБ позволяет исследовать механизмы возникновения и эволюции волновых фронтов в несжимаемой вязкой жидкости, а также взаимодействие волн, приводящее к турбулентным течениям. В ходе анализа УБ можно видеть, как при уменьшении вязкости меняется структура течения: возникают большие градиенты скорости, которые характеризуют обрушение (потерю устойчивости) волн и другие особенности течений.

Важным обстоятельством является возможность получения точных решений УБ. В работе [3] содержится их классификация и приводится более тридцати различных решений достаточно простого вида. В общем случае точное решение (1) можно найти с помощью преобразования Коула–Хопфа [4, 5], сводящего (1) к уравнению теплопроводности. Отметим однако, что в случае малых значений вязкости ν вычисление точных решений УБ по известным формулам в системах с плавающей точкой может приводить к существенным затруднениям, которые в литературе зачастую умалчиваются. Эти затруднения и способы их преодоления описаны ниже, в разделе 2.

Необходимо отметить, что для решения УБ применялись всевозможные численные методы. На раннем этапе исследований в этом направлении был предложен ряд разностных схем [6, 7]. Они сыграли значительную роль в становлении и развитии численного моделирования многих физических процессов. Позднее были предложены разностные схемы более высокого порядка аппроксимации с улучшенными свойствами устойчивости и монотонности [8, 9]. Далее в этом направлении следует отметить применение кубических B -сплайнов и других сплайнов для аппроксимации решений по пространственной переменной в комбинации со схемой Кранка–Николсон и более точными методами дискретизации

¹Это уравнение впервые получено в работе Bateman [2] за несколько десятилетий до работы Бюргерса, с которой началось его систематическое изучение.

по временной переменной [10, 11, 12], а также реализацию методов конечных элементов [13, 14] и спектральных методов [15, 16]. Для улучшения свойств точности и устойчивости использовались интегральные представления решения [17], сведение к системе ОДУ 1-го порядка с применением приближений Паде для поиска матричной экспоненты [18] и различные системы ортогональных полиномов [19, 20, 21]. Отметим также достаточно высокую гибкость и точность метода, основанного на многоуровневых вейвлетах [22]. Такие подходы позволили получить более точные численные результаты при решении УБ со значениями вязкости вплоть до $\nu \sim 10^{-4}$ с характерными значениями относительной погрешности в супремум норме: 10^{-2} – 10^{-3} , что конечно, достаточно для инженерных приложений. Однако дальнейшее уменьшение вязкости и погрешности имеет принципиальное значение для фундаментальных исследований, в частности, для анализа уравнений гидродинамики при предельно малых ν , в том числе при $\nu \rightarrow 0$. Организация таких расчётов с контролем погрешности по-прежнему представляет вызов для специалистов по численным методам.

Причина высокой сложности задачи состоит в возникновении у аналитического продолжения решения УБ особых точек, которые с течением времени движутся в комплексной плоскости. Можно показать, что при $\nu \neq 0$ все особые точки являются простыми полюсами с вычетами -2ν [23]. При уменьшении ν траектории особых точек всё ближе подходят к области действительных переменных, где ищется решение, что приводит к росту градиентов решения и его производных. Примером такой ситуации является точное решение УБ [13]

$$u(t, x) = \frac{x/t}{1 + \sqrt{t/t_0} \exp[x^2/(4\nu t)]}, \quad t \geq t_0 > 0, \quad t_0 = \exp\left(\frac{1}{8\nu}\right), \quad (2)$$

которое имеет счётное число простых полюсов в точках

$$z_k^\pm = \pm 2\sqrt{\nu t (\ln \sqrt{t_0/t} + i\pi(2k+1))}, \quad k \in \mathbb{Z}. \quad (3)$$

Нетрудно видеть, что при $t = t_0$ расстояние от особых точек z_0^\pm до действительной оси составляет $\sqrt{2\pi\nu t_0}$, что при малых ν приводит к возникновению большого градиента. Похожие эффекты реализуются для многих из известных точных решений УБ, см. решения из [3]. При этом с течением времени особая точка может как отдаляться от действительной оси (как в рассмотренном примере), обеспечивая уменьшение градиента, так и приближаться к оси, что приводит к росту градиента. Последняя ситуация характерна для задач с начальным данным, заданным целой функцией [23].

Систематическое исследование эволюции особых точек решений УБ началось в работе [24], где проведён анализ асимптотик спектров численных решений УБ, полученных методом Фурье в различные моменты времени. Для начальных данных $u(0, x) = -\sin(x)$ при $\nu = 0.05$ показано, что у решения возникает особая точка типа “полюс”, которая с ростом

t приближается к действительной оси и попадает в малую окрестность точки $x = 0$, что приводит к возникновению большого градиента решения. Однако в случае $\nu \rightarrow 0$ с ростом t особая точка “превращается” в точку ветвления типа “корень кубический” и при $t = 1$ выходит в точку $x = 0$, что приводит к разрушению классического решения. Дальнейший анализ показывает, что эта ситуация соответствует обрушению фронта волны (см., например, рис. 3 из [25]). Развитие методов и результатов из [24] содержится в [26, 27, 28, 29]. Некоторые аналитические подходы к исследованию динамики особых точек УБ при $\nu = 0$ предложены в [30].

Указанные обстоятельства позволяют сделать вывод, что аппроксимационных качеств линейных пространств полиномов (в которых, как правило, разыскивается приближённое решение УБ) оказывается недостаточно. Действительно, из прямых и обратных теорем теории приближения (см. теоремы Бернштейна, [31] и гл. 8 в [32]) следует, что для аналитических решений с особенностями в малой окрестности действительной оси скорость сходимости полиномиальных приближений будет экспоненциальной, но настолько медленной, что для достижения приемлемой точности может потребоваться сетка из многих тысяч узлов. По этой причине требуются новые методы приближения и алгоритмы для работы с ними. Поскольку у аналитического продолжения решения УБ при $\nu \neq 0$ зачастую имеется полюс, естественным выбором является применение дробно-рациональных приближений (ДРП). Отметим, что при применении ДРП для анализа дифференциальных уравнений устойчивые численные решения можно получить с использованием методов наименьших квадратов [33], которые в комбинации с методами коллокаций позволяют получить высокоточные решения с небольшим количеством коэффициентов [34].

В этой статье разработан и протестирован новый алгоритм численного решения УБ, основанный на применении дробно-рациональных барицентрических интерполяций (ДРБИ) [35, 36], метода коллокаций и метода установления. В рамках алгоритма область задачи $\{(t, x) : t \in [0, T], x \in [-1, 1]\}$ представляется в виде множества прямоугольных ячеек $[\tau_k, \tau_{k+1}] \times [-1, 1]$, где $\tau_0 = 0$, $\tau_{k+1} = \tau_k + d_k$, $k = \overline{0, K}$, $\tau_K = T$, d_k – размер шага по времени. Для поиска решения в каждой ячейке используется метод установления. При этом решение УБ приближается прямым произведением ДРБИ по x и интерполяционного полинома по t с узлами Чебышёва, см. [41], учитывающим краевые и начальные условия. Следуя идее из [37] в алгоритме используются разложения решения, полученного в предыдущей ячейке, в ряд Фурье–Чебышёва и приближения Чебышёва–Паде. Благодаря этому появляется возможность найти приближённые значения координат особой точки и адаптировать к ней пространственную сетку в каждой ячейке, а также построить траекторию движения особенности в зависимости от времени.

Заметим, что ДРБИ уже использовались для решения ряда задач с особенностями, в том числе для УБ и его обобщений [37, 38, 39, 40].

В этой работе мы впервые используем эти интерполяции в алгоритме, объединяющем итерации по времени и итерации метода установления, а также усовершенствуем метод адаптации их узлов к особенностям решения, находящимся на расстоянии $\sim 10^{-5}$ – 10^{-8} от отрезка $[-1, 1]$. Задачи с особыми точками, лежащими настолько близко к области, где ищется решение, до сих пор не рассматривались в литературе. Оказалось, что для высокоточной верификации алгоритма решения таких задач необходимы новые методы вычисления точных решений УБ. Описанию этих методов и алгоритма численного решения УБ посвящены разделы 2, 3 соответственно. Результаты численных экспериментов, сравнение с результатами других работ и обсуждения приведены в разделе 4.

2 Постановки задач для уравнения Бюргерса и вычисление точных решений

Прежде чем переходить к разработке вычислительного алгоритма, приведём постановки задач для УБ, формулы их точных решений, а также обсудим проблемы, которые возникают при использовании этих формул для расчёта на ЭВМ в системах с плавающей точкой, и способы их решения.

Для УБ (1) рассматривают либо задачу Коши (ЗК) (4) на всей числовой прямой $x \in \mathbb{R}$, либо начально-краевую задачу (НКЗ) (4), (5) на некотором отрезке (в данном случае положим $x \in [-1, 1]$):

$$u(0, x) = \varphi(x), \quad (4)$$

$$u(t, -1) = 0, \quad u(t, 1) = 0. \quad (5)$$

Решение (1) выражается с помощью преобразования Коула–Хопфа [4, 5]:

$$u(t, x) = -2\nu \frac{\partial}{\partial x} \ln F(t, x) = -2\nu \frac{F_x}{F},$$

позволяющего свести ЗК (1), (4) к задаче Коши для уравнения теплопроводности (6), (7), а НКЗ (1), (4), (5) к задаче (6)–(8):

$$F_t - \nu F_{xx} = 0, \quad t \in [0, T], \quad (6)$$

$$F(0, x) = F_0(x) = \exp \left[-\frac{1}{2\nu} \int_0^x \varphi(\xi) d\xi \right], \quad (7)$$

$$F_x(t, -1) = 0, \quad F_x(t, 1) = 0. \quad (8)$$

Решение ЗК (6), (7) на всей числовой оси даётся интегралом Пуассона

$$F(t, x) = \frac{1}{\sqrt{4\nu\pi t}} \int_{-\infty}^{\infty} \exp \left[-\frac{(x-y)^2}{4\nu t} - \frac{1}{2\nu} \int_0^y \varphi(\xi) d\xi \right] dy. \quad (9)$$

Решение НКЗ (6)–(8) разыскивается с применением метода Фурье:

$$F(t, x) = A_0 + 2 \sum_{k=1}^{\infty} A_k \cos(k\pi x) \exp(-k^2\pi^2\nu t),$$

$$A_k = \int_0^1 \cos(k\pi x) F_0(x) dx. \quad (10)$$

Решения задач (1), (4) и (1), (4), (5) выражаются соответственно по формулам

$$u_{ex}(t, x) = \frac{\int_{-\infty}^{\infty} \frac{x-y}{t} \exp\left[-\frac{(x-y)^2}{4\nu t} - \frac{1}{2\nu} \int_0^y \varphi(\xi) d\xi\right] dy}{\int_{-\infty}^{\infty} \exp\left[-\frac{(x-y)^2}{4\nu t} - \frac{1}{2\nu} \int_0^y \varphi(\xi) d\xi\right] dy}, \quad (11)$$

$$u_{ex}(t, x) = \frac{4\pi\nu \sum_{k=1}^{\infty} k A_k \sin(k\pi x) \exp(-k^2\pi^2\nu t)}{A_0 + 2 \sum_{k=1}^{\infty} A_k \cos(k\pi x) \exp(-k^2\pi^2\nu t)}. \quad (12)$$

Замечание 1. ЗК (4) и НКЗ (4), (5) имеют совершенно разный физический смысл. Тем не менее, для тестирования численных методов полезно заметить, что если ЗК поставлена так, что функции

$$\tilde{\Phi}_{\pm}(s) = \int_0^{\pm 1-s} \varphi(\xi) d\xi \text{ являются чётными функциями} \quad (13)$$

переменной s , то решение (11) удовлетворяет краевым условиям (5). В таком случае решения (11), (12) совпадают на отрезке $x \in [-1, 1]$.

Замечание 2. При использовании формулы (12) с достаточно малыми значениями t и ν нужно учитывать, что сходимость рядов в ней наблюдается при $k > 1/(\pi\sqrt{\nu t})$. Значит, для вычисления решения с высокой точностью потребуется рассчитать достаточно много членов ряда. Использование выражения для коэффициентов A_k в (10) приводит к проблеме расчёта интеграла от быстро осциллирующей функции. Наш взгляд достаточно эффективным способом поиска A_k в данном случае является применение быстрого дискретного преобразования Фурье к функции $F_0(x)$.

Отметим, что несмотря на достаточную простоту формул (11), (12), их применение при малых ν в случае начальных данных $\varphi(x)$, имеющих отрицательные значения в некоторой окрестности точки $x = 0$, приводит к проблеме. Действительно, если существует такая точка $x \in I$ (I – область решения задачи), что

$$\Phi(x) = \int_0^x \varphi(\xi) d\xi < -\Phi_M$$

где $\Phi_M = 600\nu \ln 10$, то в формулах (7), (9), (11) возникают величины, превышающие по порядку значения 10^{300} , что в системах с плавающей точкой приводит к переполнению разрядной сетки. Причина заключается в том, что при $y = x$ в формулах (9), (11) аргумент экспоненты превышает величину $\Phi_M/2\nu$, значит, значения экспоненты превышают 10^{300} . То же можно сказать и про экспоненту в формуле (7). Подобная проблема возникает, если взять $\varphi(x) = -\sin \pi x$ при $\nu \leq 0.4 \times 10^{-3}$. Начальное данное такого вида, в частности, рассматривалось в работах [16, 22, 24], в [16, 22] для поиска точных решений использованы формулы (11), (12). Однако комментарии о проблемах с расчётами по этим формулам отсутствуют.

Проблемы могут возникать и при положительных начальных данных: если $\exists x \in I : \forall y \in \mathbb{R} (x - y)^2/(2t) + \Phi(y) > \Phi_M$, то значения числителя и знаменателя (11) в такой точке x при вычислениях в системах с плавающей точкой окажутся нулевыми. Заметим также, что при расчётах в системе double изменения значений функции на величину порядка 10^{15} и более при вариации $x \in I$ нежелательны. Дело в том, что при выполнении над такими функциями операций численного дифференцирования, интегрирования и многих других возникает необходимость искать линейные комбинации их значений в различных узлах из области I . В указанном случае в силу ограниченности длины мантииссы числа double 16-ю знаками, такие операции могут привести к катастрофическому росту погрешности, связанной с округлением чисел. Для достижения высокой точности важно, чтобы изменения значений функций, стоящих в числителях и знаменателях (11), (12), при вариации $x \in I$ составляли немного порядков.

С учётом изложенного для применения формулы (11) следует наложить ограничения на начальные данные

$$\begin{aligned} \forall t > 0, x \in I \quad \exists y \in \mathbb{R} : (x - y)^2/(2t) + \Phi(y) < \Phi_m \\ \text{и } \forall x \in I \quad \Phi(x) > -\Phi_m, \end{aligned} \quad (14)$$

где $\Phi_m = 7\nu \ln 10$. При записи последнего неравенства мы полагали, что начальное данное может быть аналитически продолжено из области I в \mathbb{R} .

Если же значения ν и $\varphi(x)$ таковы, что одно из условий (14) нарушается, предлагается умножить числитель и знаменатель выражения (11) на функцию $\exp[\psi(t, x)/(2\nu)]$, где $\psi(t, x)$ задаётся таким образом, чтобы для функции

$$\Psi(t, x) = B(t, x) - \psi(t, x), \quad \text{где } B(t, x) = \min_{y \in \mathbb{R}} \left(\frac{(x - y)^2}{2t} + \Phi(y) \right), \quad (15)$$

выполнялись ограничения:

$$|\Psi(t, x)| < \Phi_m. \quad (16)$$

Выразить аналитически функцию $B(t, x)$ зачастую не удаётся, поэтому для определения $\psi(t, x)$ можно использовать системы символьных

вычислений, способные преодолеть ограничения систем с плавающей точкой. Опишем способ вычисления функции $\psi(t, x)$ на примере задачи с начальным данным $\varphi(x) = -\sin(\pi x)$, $x \in [-1, 1]$. Для простоты изложения фиксируем $\nu = 10^{-4}$, $t = 1$. Тогда функция $\psi(x) := \psi(1, x)$ может быть выражена формулой

$$\psi(x) = -(2\nu M \ln 10) \bar{\psi}_P(x), \quad (17)$$

где величина M описана ниже,

$$\bar{\psi}_P(x) = -0.59(x - \text{sign}(x))^2 + 1.$$

Поясним, как получена $\psi(x)$. Обозначим $D_p(x)$ – значение десятичного логарифма знаменателя (11); $\bar{D}_p(x) = D_p(x)/M$, где $M = \|D_p(x)\| = D_p(1)$, $\|\cdot\|$ – супремум норма функции на отрезке $[-1, 1]$. Десятичный логарифм знаменателя (11) после умножения на корректирующую функцию $\exp(\psi(x)/(2\nu))$ обозначим

$$D_p^{\text{cor}}(x) = D_p(x) + \psi(x)/(2\nu \ln 10).$$

На рис. 1, а изображён график функции $D_p(x)$, из которого видно, что значения знаменателя (11) существенно превышают величину 10^{300} и разброс этих значений при $x \in I$ превышает 700 порядков. Такая ситуация абсолютно непригодна для организации вычислений в системах с плавающей точкой.

Аппроксимация нормированной функции $\bar{D}_p(x)$ полиномом $\bar{\psi}_P(x)$ (см. рис 1, б) позволяет рассчитать $\psi(x)$. Действительно, пусть относительная погрешность указанной аппроксимации есть

$$\varepsilon_P = \|\bar{D}_p(x) - \bar{\psi}_P(x)\| / \|\bar{D}_p(x)\| \ll 1.$$

Тогда, если также как в (17), взять $\psi(x) = -(2\nu M \ln 10) \bar{\psi}_P(x)$, то

$$\begin{aligned} \|D_p^{\text{cor}}(x)\| &= \|D_p(x) - M\bar{\psi}_P(x)\| = M\|\bar{D}_p(x) - \bar{\psi}_P(x)\| \leq \\ &\leq \varepsilon_P M \max_{x \in [-1, 1]} |\bar{D}_p(x)| = \varepsilon_P \max_{x \in [-1, 1]} |M\bar{D}_p(x)| = \varepsilon_P \|D_p(x)\| \ll \|D_p(x)\|. \end{aligned}$$

График $D_p^{\text{cor}}(x)$ изображён на рис. 1, в.

Отметим, что введение поправочного множителя $\exp(\psi(x)/(2\nu))$ обеспечивает реализацию ограничений (16). На рис. 2 приведены графики функций $B(1, x)$ и $\Psi(1, x)$. Из графиков видно, что второе из ограничений (14) для функции $B(1, x)$ нарушается, в то время как для функции $\Psi(1, x)$ ограничения (16) выполнены со значительным запасом (в данном примере $\Phi_m \approx 0.0035$). Кроме того, указанный множитель «балансирует» значения числителя и знаменателя (11) так, что разброс их порядков при вариации $x \in I$ существенно меньше 15-ти: в данном эксперименте разброс значений составляет примерно $10^{-2} - 10^2$, рис. 1, в, то есть примерно равен четырём порядкам.

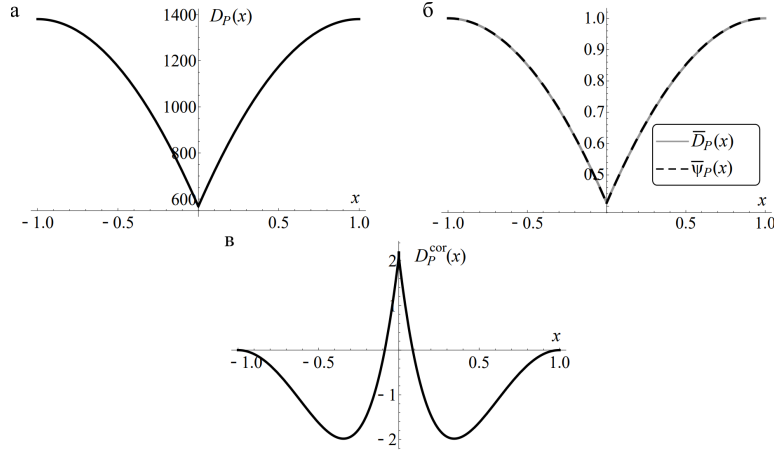


Рис. 1. Знаменатель решения (11) при $\varphi(x) = -\sin \pi x$ и $\nu = 10^{-4}$ и корректировка его значений: графики функций $D_P(x)$ (а); $\bar{D}_P(x)$ и её аппроксимации $\bar{\psi}_P(x)$ (б); функции $D_P^{\text{cor}}(x)$, полученной при использовании корректирующего множителя (в).

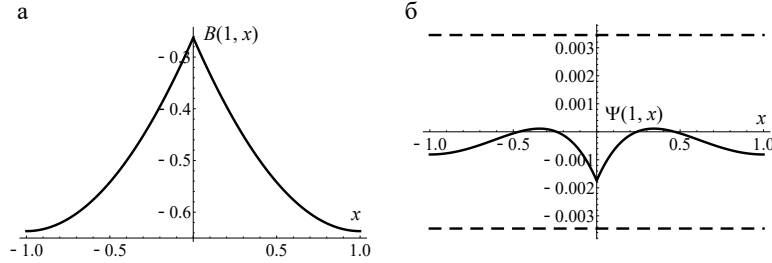


Рис. 2. Корректировка аргументов экспонент в (11) при $\varphi(x) = -\sin \pi x$ и $\nu = 10^{-4}$: графики функции $B(1, x)$ (а) и функции $\Psi(1, x)$ – ограничения $\pm\Phi_m$ из неравенства (16) показаны пунктиром (б).

В общем случае при $\varphi = -\sin \pi x$, $t > 0$ и сколь угодно малых ν для приближения функции $\bar{D}_P(x)$ будем использовать аппроксимации вида

$$\bar{\psi}_P(t, x) = \sum_{k=0}^{\bar{K}} A_{\psi, k} T_k(2|x| - 1),$$

где T_k – многочлен Чебышёва первого рода степени k . Такой вид аппроксимации учитывает чётность функции, стоящей в знаменателе (11) (её легко показать, используя чётность $\Phi(y)$). Далее снова положим $\psi(x) = -(2\nu M \ln 10) \bar{\psi}_P(x)$. Экспериментально установлено, что для выполнения условий (16) для функции $\Psi(t, x)$ достаточно задать число членов ряда \bar{K} таким образом, чтобы выполнялось неравенство: $|A_{\psi, k}| < 10\nu$.

Для вычисления коэффициентов $A_{\psi,k}$ предлагается использовать следующий код на языке Wolfram Mathematica:

```

nu := 10^(-5);
t := 0.5;
Nn0 := 3;
coefs = ConstantArray[1, Nn0-2];
Nn = Nn0;
DP[x_] := Quiet[NIntegrate[Exp[-(x-y)^2/(4*nu*t)] - (Cos[Pi*y]-1)/(2*nu*Pi)],
{y, -Infinity, Infinity}, Method -> "ExtrapolatingOscillatory",
WorkingPrecision -> 35, PrecisionGoal -> 15, MaxRecursion -> 20];
M := Log10[DP[1]];
DDP[x_] := Log10[DP[x]]/M;
While[Abs[Last[coefs]] > nu*10,
  Nn = Nn+2;
  NodesF[j_] := (Cos[(2*j-1)*Pi/(2*Nn)]+1)/2;
  Nodes := Array[NodesF, Nn, {1, Nn}];
  BaseF[j_] := ChebyshevT[j, 2*x-1];
  Base := Array[BaseF, Nn, {0, Nn-1}];
  DDPL := Transpose[{Nodes, DDP[Nodes]}];
  FFcor[x_] = Fit[DDPL, Base, x];
  coef[j_] := Coefficient[FFcor[x], ChebyshevT[j, 2*x-1], 1];
  coefs = Array[coef, Nn-2, {2, Nn-1}];
]
ress[x_] := FFcor[x] - Sum[coefs[[j-1]]*ChebyshevT[j, 2*x-1], {j, 2, Nn-1}];
coefs = Join[{Coefficient[ress[x], ChebyshevT[1, -1+2*x], 0],
  Coefficient[ress[x], ChebyshevT[1, -1+2*x], 1]}, coefs];
Apsi = -2*nu*M*Log[10]*coefs;
Print["Number of coefficients of expansion over ", T_k, " is
overline{K}+1=", Nn]
Print["The coefficients are \n", Apsi]

(*You can save the coefficients to file*)
(*Export["c:/coefs.txt", Apsi, "TSV"];*)

(*You can plot the error of approximation of \overline{D}_P by
\overline{\psi}(t,x). This can take a time*)
(*
Plot[DDP[x]-FFcor[x], {x,0,1}, PlotStyle->{Thickness[0.007], Black},
LabelStyle -> {FontFamily -> "Times New Roman", 20, GrayLevel[0]},
ImageSize -> Large, PlotLegends -> Placed[LineLegend[{"Error of
approximation of DP by psi(t,x)"}], LegendFunction -> Frame,
LegendMarkerSize -> {{55, 10}}, {0.5, 0.2}]]*)

(*You can plot the value of \Psi(t,x). This can take a time*)
(*
FFcor2[x_] := Sum[Apsi[[j]]*ChebyshevT[j-1, 2*Abs[x]-1], {j, 1, Nn}];
Psi[x_] := MaxValue[-(x-y)^2/(4*nu*t) - (Cos[Pi*y]-1)/(2*nu*Pi) +
FFcor2[x]/(2*nu), y];
Plot[Log10[Abs[Psi[x]]], {x, -1, 1}, PlotStyle->{Thickness[0.007], Black},
LabelStyle -> {FontFamily -> "Times New Roman", 28, GrayLevel[0]},
ImageSize -> Large, PlotPoints -> 200]*)

```

В данной работе рассматривается начально-краевая задача для УБ

$$\begin{cases} u_t + uu_x = \nu u_{xx}, & x \in [-1, 1], \quad t \in [0, T] \\ u(0, x) = \varphi(x) = -\sin \pi x, \\ u(t, -1) = 0, \quad u(t, 1) = 0. \end{cases} \quad (18)$$

На рис. 3 представлен график численного решения задачи (18), полученного в расчётах по алгоритму, описанному ниже.

В силу замечания 1 точное решение НКЗ (18) может быть выражено формулой (11).

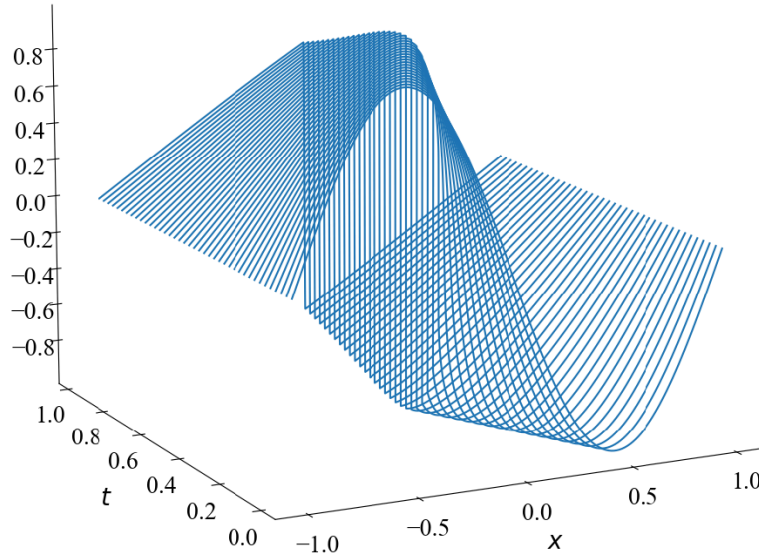


Рис. 3. График численного решения задачи (18) при $\nu = 10^{-4}$, $T = 1$.

3 Алгоритм на основе барицентрической интерполяции

В этом разделе описаны интерполяционные формулы, служащие для приближения решения задачи (18) и приведены выражения для элементов матриц, аппроксимирующих операторы дифференцирования по переменным x и t в этой задаче. Далее для поиска решений разработан вычислительный алгоритм на основе метода установления с использованием методов коллокации и добавочной функции. Суть метода коллокации заключается в подстановке в уравнение приближения решения и записи уравнения в наборе узлов коллокации. Метод добавочной функции используется для того, чтобы при аппроксимации решения сразу учесть начальные данные.

Для приближения решения по переменной $x \in [-1, 1]$ при фиксированном t за основу возьмём дробно-рациональную барицентрическую интерполяционную формулу

$$u(t, x) \approx R_N(x) = \frac{\sum_{j=1}^N \frac{\omega_j u(t, g(x_j^{\text{ch}}))}{x - g(x_j^{\text{ch}})}}{\sum_{j=1}^N \frac{\omega_j}{x - g(x_j^{\text{ch}})}}, \quad (19)$$

где $x_j^{\text{ch}} = \cos \frac{(2j-1)\pi}{2N}$, $\omega_j = (-1)^{j-1} \sin \frac{(2j-1)\pi}{2N}$ – узлы и веса интерполяции соответственно, $j = \overline{1, N}$, $g(x)$ – конформное отображение, переводящее отрезок $[-1, 1]$ в себя и позволяющее адаптировать положение узлов к особой точке, $g(\pm 1) = \pm 1$. В дальнейшем положим $x_j = g(x_j^{\text{ch}})$. Отметим, что приближение (19) было предложено в [35], оценки его погрешности получены в [36]. Эффективность применения (19) при решении УБ показана в [38, 37],

Решение задачи (18) в области $C_k = \{(t, x) : t \in [\tau_k, \tau_{k+1}], x \in [-1, 1]\}$, где отрезок $[\tau_k, \tau_{k+1}]$ будет описан ниже, ищем в виде прямого произведения следующих интерполяционных формул:

- Модифицированная интерполяция (19) с фиксированным $t \in [\tau_k, \tau_{k+1}]$:

$$\widehat{R}_N(x) = \sum_{j=1}^N \frac{1-x^2}{1-x_j^2} \frac{\omega_j u(t, x_j)}{x-x_j} \bigg/ \sum_{j=1}^N \frac{\omega_j}{x-x_j}, \quad x_j = g(x_j^{\text{ch}}). \quad (20)$$

- Модифицированный интерполяционный полином с узлами Чебышёва, $\theta_i = \cos \frac{(2i-1)\pi}{2M}$, и весами $\xi_i = (-1)^{i-1} \sin \frac{(2i-1)\pi}{2M}$, $i = \overline{1, M}$, при $t \in [\tau_k, \tau_{k+1}]$ с фиксированным $x \in [-1, 1]$:

$$\widehat{P}_M(t) = \frac{1}{M} \sum_{i=1}^M \frac{1+\Lambda(t)}{1+\theta_i} \frac{\xi_i T_M(\Lambda(t))}{\Lambda(t)-\theta_i} u(t_i, x) + u(\tau_k, x), \quad (21)$$

где $\Lambda(t)$ – линейное отображение отрезка $[\tau_k, \tau_{k+1}]$ в отрезок $[-1, 1]$, $t_i = \Lambda^{-1}(\theta_i)$, T_M – полином Чебышёва степени M .

В случае функции (20) модификация интерполяции заключается в наличии множителя $\frac{1-x^2}{1-x_j^2}$, $x_j = g(x_j^{\text{ch}})$, что позволяет учесть нулевые краевые условия, а в случае функции (21) – в наличии множителя $\frac{1+\Lambda(t)}{1+\theta_i}$ и

слагаемого $u(\tau_k, x)$, что в соответствии с методом добавочной функции обеспечивает реализацию начального условия.

Замечание 3. Введение указанных множителей играет принципиальную роль в создании алгоритма на основе метода коллокаций. Действительно, непосредственное применение (19) приводит к задаче линейной алгебры, в которой матрицы, аппроксимирующие производные (см. (23), матрицы $D_{(1)}$ и $D_{(2)}$), будут иметь только нулевые собственные числа и Жордановы клетки большого размера, соответствующие этим числам. Это является прямым следствием того, что операторы дифференцирования первого и второго порядков в пространстве полиномов имеют только постоянные и линейные собственные функции и им соответствуют нулевые собственные числа. Описанная ситуация абсолютно непригодна для организации вычислений.

Замечание 4. *Приближение (20) в случае, когда $g(x)$ – тождественное отображение, является полиномом, в противном случае (20) – это дробно-рациональная функция (ДРФ). Алгоритм адаптации пространственной сетки к особой точке, основанный на использовании специального вида $g(x)$, описан ниже.*

Ввиду применения (21) в каждой области C_k сетка по времени задана узлами Чебышёва, θ_i , $i = \overline{1, M}$, которые сгущены возле границ, но не включают в себя сами границы. Предложенный численный алгоритм, использующий метод коллокации, позволяет находить приближенные значения решения в данных узлах. Однако для реализации метода добавочной функции в области C_{k+1} нам потребуется вычислить значение решения на правой границе рассмотренного отрезка по времени $t = \tau_{k+1}$. Для этого воспользуемся интерполяционным полиномом с узлами Чебышёва (без модификации):

$$u(\tau_{k+1}, x) \approx P_M(\tau_{k+1}) = \frac{1}{M} \sum_{i=1}^M \frac{\xi_i}{1 - \theta_i} u(t_i, x), \quad (22)$$

где $x \in [-1, 1]$ фиксировано.

Для получения матриц, аппроксимирующих производные по t и x , необходимо продифференцировать используемые интерполяционные формулы нужное число раз, а после перейти к пределам $t \rightarrow t_i$ и $x \rightarrow x_j$. Для приближения (19) элементы матриц $D_{(1)}$ и $D_{(2)}$, аппроксимирующих первую и вторую производные, приведены в [37]:

$$D_{(1)ij} = \begin{cases} \frac{\omega_j}{\omega_i(x_i - x_j)}, & \text{при } i \neq j; \\ -\chi_i, & \text{при } i = j; \end{cases} \quad (23)$$

$$D_{(2)ij} = \begin{cases} \gamma_{ij}, & \text{при } i \neq j; \\ -\sum_{k=1, k \neq i}^N \gamma_{ik}, & \text{при } i = j; \end{cases}$$

где $\chi_i = \sum_{k=1, k \neq i}^N \frac{\omega_k}{\omega_i(x_i - x_k)}$, $\gamma_{ik} = -\frac{2\omega_k}{\omega_i(x_i - x_k)} \left(\chi_i + \frac{1}{x_i - x_k} \right)$.

Далее приведены матричные аппроксимации операторов дифференцирования, полученные из модифицированных интерполяционных формул:

- $N \times N$ - матрица $B_{(1)} = (B_{(1)ij})$, аппроксимирующая первую производную по x , полученная из (20):

$$B_{(1)ij} = \begin{cases} \frac{1 - x_i^2}{1 - x_j^2} \frac{\omega_j}{\omega_i(x_i - x_j)}, & \text{если } i \neq j; \\ -\frac{2x_i}{1 - x_i^2} - \chi_i, & \text{если } i = j. \end{cases} \quad (24)$$

- $N \times N$ - матрица $B_{(2)} = (B_{(2)ij})$, аппроксимирующая вторую производную по x , полученная из (20):

$$B_{(2)ij} = \begin{cases} -\frac{4x_i}{1-x_j^2} \frac{\omega_j}{\omega_i(x_i-x_j)} + \frac{1-x_i^2}{1-x_j^2} \gamma_{ij}, & \text{если } i \neq j; \\ -\frac{2}{1-x_i^2} + \frac{4x_i}{1-x_i^2} \chi_i - \sum_{k=1, k \neq i}^N \gamma_{ik}, & \text{если } i = j. \end{cases} \quad (25)$$

- $M \times M$ - матрица $A = A_{ij}$, аппроксимирующая первую производную по t , полученная из (21):

$$A_{ij} = \begin{cases} (-1)^{i+j} \frac{2}{\tau_{k+1} - \tau_k} \frac{(1+\theta_i)\sqrt{1-\theta_j^2}}{(1+\theta_j)(\theta_i-\theta_j)\sqrt{1-\theta_i^2}}, & \text{если } i \neq j; \\ \frac{2}{\tau_{k+1} - \tau_k} \left(\frac{1}{1+\theta_i} + \frac{\theta_i}{2(1-\theta_i^2)} \right), & \text{если } i = j. \end{cases} \quad (26)$$

Область определения задачи (18), $\hat{\Omega} = [0, T] \times [-1, 1]$, разобьем на ячейки следующим образом: введем шаги d_k и сетку с узлами τ_k , такую что $\tau_0 = 0$, $\tau_{k+1} = \tau_k + d_k$, $\tau_K = T$, $k = \overline{0, K}$.

В каждой из ячеек $C_k = [\tau_k, \tau_{k+1}] \times [-1, 1]$ ищем решение с применением метода коллокации, метода установления и базисной функции в виде прямого произведения (20) и (21). Для этого в C_k введём сетку (t_i, x_j) , где $t_i = \Lambda^{-1}(\theta_i)$, $\theta_i = \cos \frac{(2i-1)\pi}{2M}$, $i = \overline{1, M}$. В ячейке C_0 положим $x_j = x_j^{\text{ch}} = \cos \frac{(2j-1)\pi}{2N}$, $j = \overline{1, N}$, а в следующих ячейках, $x_j = g(x_j^{\text{ch}})$. Функцию $g(x)$ будем строить так, чтобы адаптировать сетку к особенности решения.

Для реализации метода установления при вычислении решения в каждой ячейке C_k введем дополнительную переменную s и положим $u = u_0(t, x, s) + u_{C_k}(x)$, где u_0 удовлетворяет однородному начальному условию в точке $t = \tau_k$, $u_{C_k}(x)$ – добавочная функция, отвечающая за начальное условие в ячейке C_k , которая рассчитывается по значениям решения из предыдущей ячейки с применением (22). Будем искать $u_0(t, x, s)$ как предел решения задачи:

$$\frac{\partial u_0}{\partial s} = u_{0t} + (u_0 + u_{C_k})(u_0 + u_{C_k})_x - \nu(u_0 + u_{C_k})_{xx} \text{ при } s \rightarrow \infty, \quad (27)$$

полагая $u(t, x, 0) = 0$. Введем равномерную сетку по s с шагом Δ и узлами $s^n = n\Delta$, $n = 1, 2, \dots$. Обозначим $u_0^n = u_0(t, x, s^n)$, тогда $\frac{\partial u_0}{\partial s} \approx \frac{u_0^{n+1} - u_0^n}{\Delta}$, и схема метода установления для решения в ячейке C_k имеет вид:

$$\frac{u_0^{n+1} - u_0^n}{\Delta} = u_{0t}^{n+1} + (u_0^n + u_{C_k})(u_0^n + u_{C_k})_x - \nu(u_0^{n+1} + u_{C_k})_{xx}. \quad (28)$$

Применим метод коллокации. Пусть

- $U_0^n - M \times N$ - матрица, содержащая приближение значений решения на n -ой итерации метода установления в узлах коллокации, M - количество узлов по времени и N - по пространству.
- $U_{C_k} - M \times N$ - матрица, строки которой являются тождественными и содержат значения решения в узлах $x_j, j = \overline{1, N}$, при $t = \tau_k$, полученные при использовании решения в предыдущей ячейке.

Запишем приближённые равенства:

- $(U_0^n + U_{C_k})_x \approx (U_0^n + U_{C_k})B_{(1)}^T$, где элементы $B_{(1)}$ определены в (24).
- $(U_0^{n+1} + U_{C_k})_{xx} \approx (U_0^{n+1} + U_{C_k})B_{(2)}^T$, где элементы $B_{(2)}$ заданы в (25).
- $U_{0t}^{n+1} \approx AU_0^{n+1}$, где элементы A заданы в (26).

Здесь нижние индексы x, xx и t у матриц означают, что они содержат значения соответствующих производных в узлах коллокации.

Записывая (28) в узлах коллокации, получим матричное уравнение:

$$U_0^{n+1} - \Delta AU_0^{n+1} + \Delta \nu U_0^{n+1} B_{(2)}^T = U_0^n + \Delta \left[(U_0^n + U_{C_k}) \odot ((U_0^n + U_{C_k})B_{(1)}^T) - \nu U_{C_k} B_{(2)}^T \right], \quad (29)$$

где символ " \odot " обозначает поэлементное произведение матриц.

Далее используем спектральное разложение матриц A и $B_{(2)}^T$:

$$A = R_A D_A R_A^{-1}, \quad B_{(2)}^T = R_B D_B R_B^{-1}, \quad (30)$$

где R_A, R_B - матрицы, состоящие из собственных векторов, а D_A, D_B - диагональные матрицы собственных значений A и $B_{(2)}^T$. Матрицы A и $B_{(2)}^T$ допускают спектральное разложение (30) в силу того, что кратность их собственных чисел равна единице, данное свойство подробно исследовано в [42, 43].

Умножаем уравнение (29) слева на R_A^{-1} и справа на R_B . Введем обозначение: $V^{n+1} = R_A^{-1} U_0^{n+1} R_B$, после чего уравнение (29) примет вид:

$$V^{n+1} - \Delta D_A V^{n+1} + \Delta \nu V^{n+1} D_B = R_A^{-1} \left(U_0^n + \Delta \left[(U_0^n + U_{C_k}) \odot ((U_0^n + U_{C_k})B_{(1)}^T) \right] \right) R_B - \nu \Delta R_A^{-1} U_{C_k} R_B D_B. \quad (31)$$

обозначим как G^{n+1}

Выразим элементы матрицы V^{n+1} из (31):

$$V_{ij}^{n+1} = G_{ij}^{n+1} / (1 - \Delta D_{A,i} + \Delta \nu D_{B,j}), \quad (32)$$

где $D_{A,i}$ - i -ое собственное число матрицы A , $D_{B,j}$ - j -ое собственное число $B_{(2)}^T$, G_{ij}^{n+1} - значение правой части (31) в узле (t_i, x_j) . Решение на следующем шаге метода установления выразим в виде $U_0^{n+1} = R_A V^{n+1} R_B^{-1}$, увеличим n на единицу, и повторим описанные вычисления.

Будем проводить итерации метода установления в данной ячейке, пока решение не установится. Критерий остановки:

$$\max |U_0^{n+1} - U_0^n| \leq \varepsilon_S \Delta, \quad (33)$$

где ε_S – невязка установления, малое число. Здесь максимум берётся по всем элементам матрицы. В качестве U_0^0 зададим нулевую матрицу. В случае невыполнения критерия остановки в текущей ячейке при $n = \overline{1, 199}$ будем уменьшать ее размер в два раза: $d_k = d_k/2$ и действовать таким образом до тех пор, пока не добьёмся сходимости метода установления в ячейке C_k при числе итераций меньшем 200.

Теперь, используя (22), получим численное решение $u(\tau_{k+1}, x)$ и отталкиваясь от идей из работы [37], опишем метод локализации особой точки решения $z^*(t) = \delta(t) + i\varepsilon(t) \in \mathbb{C}$ и адаптации к ней пространственной сетки x_j , $j = \overline{1, N}$. В рамках этого метода мы полагаем, что у искомого решения существует изолированная особая точка, лежащая достаточно близко к области задачи. Для поиска приближенных значений $\delta(\tau_{k+1})$ и $\varepsilon(\tau_{k+1})$ разложим $u(\tau_{k+1}, x)$ в ряд Фурье–Чебышёва и аппроксимируем его приближением Чебышёва–Паде:

$$u(\tau_{k+1}, x) = \sum_{k=0}^{\infty} a_k T_k(x) \approx \frac{\sum_{k=0}^m b_k T_k(x)}{1 + \sum_{k=1}^r c_k T_k(x)} = \frac{p(x)}{q(x)}. \quad (34)$$

Отсюда, требуя, чтобы $u(\tau_{k+1}, x)q(x) - p(x) = O(T_{m+r+1}(x))$, получаем систему уравнений:

$$\begin{pmatrix} 1 & & -d_{0,1} & \cdots & -d_{0,r} \\ & \ddots & \vdots & \vdots & \vdots \\ & & 1 & -d_{m,1} & \cdots & -d_{m,r} \\ & & & -d_{m+1,1} & \cdots & -d_{m+1,r} \\ & & & \vdots & \vdots & \vdots \\ & & & -d_{m+r,1} & \cdots & -d_{m+r,r} \end{pmatrix} \begin{pmatrix} b_0 \\ \vdots \\ b_m \\ c_1 \\ \vdots \\ c_r \end{pmatrix} = \begin{pmatrix} a_0 \\ \vdots \\ a_m \\ a_{m+1} \\ \vdots \\ a_{m+r} \end{pmatrix}.$$

Коэффициенты $d_{i,j}$ находятся из условия $\sum_{i=0}^{\infty} d_{i,j} T_i(x) = \sum_{i=0}^{\infty} a_i T_i(x) T_j(x)$ с учётом формулы $T_i(x) T_j(x) = \frac{1}{2}(T_{i+j}(x) - T_{|i-j|}(x))$. Для локализации особенности требуется найти c_i , $i = \overline{1, r}$. Поскольку последние r уравнений системы, записанной выше, не включают b_i , $i = \overline{0, m}$, необходимо решать только подсистему для коэффициентов c_i . Возьмем $r = 2$, тогда эти коэффициенты можно выразить явно [39]:

$$\begin{aligned} d_{m+1,1} &= a_{m+1}/2 + a_{m+3}/2, & d_{m+1,2} &= a_m/2 + a_{m+4}/2, \\ d_{m+2,1} &= a_{m+2}/2 + a_{m+4}/2, & d_{m+2,2} &= a_{m+1}/2 + a_{m+5}/2. \end{aligned}$$

После решения подсистемы необходимо отыскать корни знаменателя в приближении (34). Если корнями являются два сопряженных комплексных числа и мнимая часть больше 10^{-14} , то $\delta(\tau_{k+1})$ и $\varepsilon(\tau_{k+1})$ – действительная и мнимая часть одного из корней. Без ограничения общности выберем корень с положительным $\varepsilon(\tau_{k+1})$. Если $|\varepsilon(\tau_{k+1})| \leq 10^{-14}$ или

модули значений $\delta(\tau_{k+1})$ и $\varepsilon(\tau_{k+1})$ велики, то полагаем $\delta(\tau_{k+1}) = \delta(\tau_k)$ и $\varepsilon(\tau_{k+1}) = \varepsilon(\tau_k)$.

Для нахождения коэффициентов a_k в (34) можно вычислить $u(\tau_{k+1}, x)$ в узлах Чебышёва на отрезке $[-1, 1]$, а после применить быстрое преобразование Фурье. Однако если известно положение особенности на предыдущем шаге по времени τ_k , то для более точного расчёта $\varepsilon(\tau_{k+1})$, $\delta(\tau_{k+1})$ приближение (34) строится на отрезке $[\max(-1, \delta(\tau_k) - 10\varepsilon(\tau_k)), \min(1, \delta(\tau_k) + 10\varepsilon(\tau_k))]$.

Наконец, когда получены $\delta(\tau_{k+1})$ и $\varepsilon(\tau_{k+1})$, можно определить новую сетку по пространству для следующей ячейки C_{k+1} . Для этого используется отображение $g(x)$, которое конформно переводит отрезок $[-1, 1]$ в себя, см. [37]:

$$\hat{x}_j = g(x_j) = \delta(\tau_{k+1}) + \varepsilon(\tau_{k+1}) \sinh \left[(a^- + a^+) \frac{x_j - 1}{2} + a^- \right], \quad (35)$$

где $a^\pm = \sinh^{-1} \frac{1 \pm \delta(\tau_{k+1})}{\varepsilon(\tau_{k+1})}$, $j = \overline{1, N}$.

Важным обстоятельством является то, что найти численное решение с применением (35) при $\nu < 10^{-5}$ не удавалось. В связи с этим разработаем модификацию (35). Анализ решения (18) показал, что действительная координата особой точки может быть с высокой точностью аппроксимирована нулём, см. рис. 6, поэтому зафиксируем $\delta(t) = 0$ и вместо $g(x)$ применим композицию $g \circ g$:

$$\hat{x}_j = g(g(x_j)) = \varepsilon(\tau_{k+1}) \sinh \left[\frac{\pi}{2} \sinh \left(x_j S(\tau_{k+1}) \right) \right], \quad j = \overline{1, N}, \quad (36)$$

где $S(\tau_{k+1}) = \sinh^{-1} \left(\frac{2}{\pi} \sinh^{-1} \frac{1}{\varepsilon(\tau_{k+1})} \right)$. Указанная композиция позволяет «унести» особую точку $z^*(t)$ дальше от отрезка $[-1, 1]$ в \mathbb{C} , чем исходное отображение g . Применение $g \circ g$ при $\nu \geq 10^{-5}$ позволило существенно улучшить устойчивость работы алгоритма и получить более высокую точность при фиксированном числе узлов.

4 Численные результаты

Ниже представлены результаты исследования точности и скорости работы созданного метода при решении начально-краевой задачи для УБ (18). Для верификации полученных результатов в п. 4.1 описан метод расчёта точного решения по формуле (11) и исследована его сходимость. В п. 4.2 показано, что построение адаптивных методов и обоснованный выбор адаптации являются исключительно важными уже при $\nu = 0.01$ и приведено обсуждение нюансов реализации предложенного метода и полученных решений. Наконец, в п. 4.3 проведено сравнение этих решений с решениями других авторов (сравнение значений производной $\partial u / \partial x$ при $x = 0$). Для оценки качества результатов вычислено точное значение этой производной с применением формулы (11). Отметим сразу,

что в рассмотренной задаче решения при $\nu < 10^{-3}$ в открытых источниках обнаружены не были. Нам же удалось рассчитать решения при предельно малых ν вплоть до $\nu = 10^{-8}$.

4.1. Расчёт точного решения. Для вычисления точного решения задачи (18) рассчитаем интегралы, стоящие в формуле (11), в момент времени $t = T$ с применением квадратурной формулы "двойной экспоненты", описанной в работе [44]:

$$I_h = \frac{\pi}{2} h \sum_{n=-H}^H w\left(t, x, \sinh\left(\frac{\pi}{2} \sinh nh\right)\right) (\cosh nh) \cosh\left(\frac{\pi}{2} \sinh nh\right), \quad (37)$$

где $h = \sinh^{-1}\left(\frac{2}{\pi} \sinh^{-1}(10^{19})\right) / H$, а $2H+1$ – количество узлов квадратурной формулы, $w(t, x, y)$ – подынтегральная функция. Использование числа 10^{19} в выражении для шага h служит численной аппроксимацией бесконечности. При расчётах по формуле (11) с начальным данным $\varphi(x) = -\sin \pi x$ и вязкостью $\nu < 10^{-3}$ используем поправочный множитель с функцией $\psi(t, x)$, рассчитанной как описано в разд. 2.

На рис. 4 изображена сходимость приближений точного решения (11) при расчёте интегралов, стоящих в нём, по формуле (37) при различных ν – показан график $d_H = \lg\left(\frac{\|u_{ex,H} - u_{ex,2H}\|_\infty}{\|u_{ex,2H}\|_\infty}\right)$ в зависимости от H при $T = 0.5$. Можно сделать вывод, что при достаточно больших H решение u_{ex} рассчитывается с высокой точностью. Отметим, что при меньших значениях ν такие графики мы не строили, однако всегда проверяли, что при увеличении значения H значение относительной погрешности err (см. формулу ниже) меняется не более, чем на 1%.

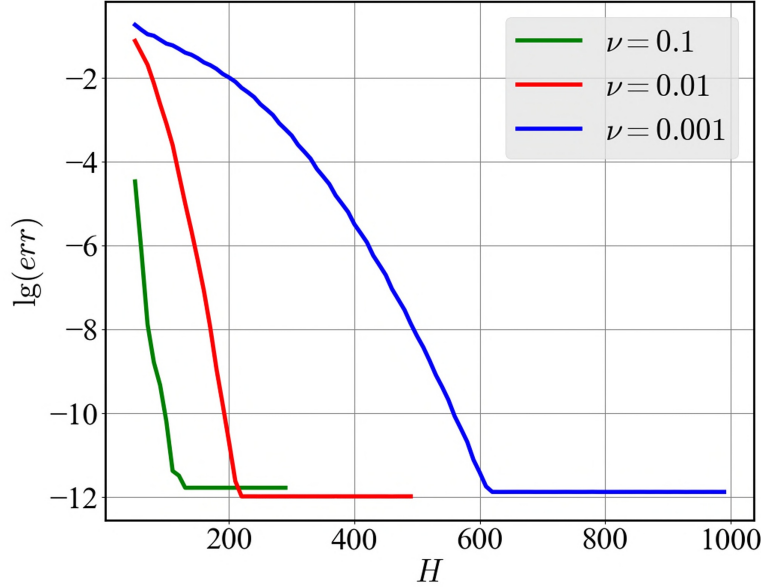
4.2. Результаты работы предложенного метода. Для исследования точности расчётов зададим равномерную сетку по x из N_p узлов: $\check{x} = \{x_j : x_j = -1 + 2(j-1)/(N_p-1), j = \overline{1, N_p}\}$ и введём обозначение относительной погрешности решения задачи (18):

$$err = \frac{\|u(T, x) - u_{ex}(T, x)\|_\infty}{\|u_{ex}(T, x)\|_\infty},$$

где $\|\cdot\|_\infty = \max_{x_j \in \check{x}} |\cdot|$. Если при описании эксперимента значение N_p не указано, то положим $N_p = 10000$.

Для более детального исследования точности численного решения задачи (18) определим также локальную погрешность

$$err_{loc} = \frac{\|u(T, x) - u_{ex}(T, x)\|_{\infty, loc}}{\|u_{ex}(T, x)\|_{\infty, loc}},$$


 Рис. 4. График d_H в зависимости от H при $T = 0.5$.

где $\|\cdot\|_{\infty,loc} = \max_{x_j \in \bar{x}} |\cdot|$, для подсчета которой вводится равномерная сетка $\bar{x} = \{x_j : x_j = -100\nu + 200\nu(j-1)/(N_p-1), j = \overline{1, N_{p,loc}}\}$, $x_j \in [-100\nu, 100\nu]$.

Далее опишем основные параметры численного метода и укажем их значения, использованные в расчётах:

1. N – число узлов пространственной сетки, $N \in [100, 350]$.
2. K – число ячеек, на которые разбита область задачи, может быть фиксировано или изменяться в процессе вычислений при адаптации алгоритма, $K \in [10, 2 \times 10^6]$.
3. M – число узлов Чебышёва сетки по времени в каждой ячейке, должно быть достаточно большим для обеспечения высокой точности приближения (21), но не слишком большим для уменьшения вычислительных затрат и сохранения численной устойчивости алгоритма. Экспериментально установлено, что $M = 10$ позволяет получить наименьшие err .
4. Δ – шаг метода установления, $\Delta = 1$. Такое значение обеспечивает высокую скорость сходимости метода в большинстве расчётных режимов.
5. ε_S – невязка установления, см. (33). Если не указано иного, $\varepsilon_S = 10^{-12}$.
6. H – параметр квадратурной формулы (37) при вычислении точного решения (11), $H \in [300, 10^5]$.

7. $N_p, N_{p,loc}$ – число узлов равномерной сетки по x , на которой рассчитывается относительная погрешность, $N_p \in [10^4, 10^5]$, а также локальная погрешность, $N_{p,loc} \in [5 \cdot 10^3, 2 \cdot 10^5]$. Если не указано иного, то $N_p = 10^4$.

Использование ДРП по пространственной переменной и адаптация сетки к положению особенности позволяют получать высокоточное численное решение при малых ν . На рис. 5 представлены графики зависимости $\lg(err)$ от N при $T = 0.5$, $\nu = 0.1, 0.01$, для решений, полученных алгоритмом с адаптацией пространственной сетки к особой точке (сплошные линии) и без адаптации (пунктир). В этих экспериментах для адаптации сетки к особенности отображение g применялось однократно. Сплошные линии на представленном графике показывают быстрое экспоненциальное убывание ошибки при достаточно малых N . Для достижения сопоставимой точности с помощью алгоритма без адаптации требуется существенно большее число узлов N , что характеризует эффективность адаптивного метода. Его эффективность также подтверждается тем, что алгоритм без адаптации сетки в задачах с вязкостью $\nu < 10^{-2}$ не сходится. Адаптации с однократным применением отображения g позволила получить численное решение при $\nu \gtrsim 10^{-5}$, а с применением композиции $g \circ g$ – при значениях ν вплоть до 10^{-8} .

При тестировании адаптивного алгоритма было проведено множество численных экспериментов, результаты которых суммированы в табл. 1, 2. При получении этих результатов везде использована композиция $g \circ g$, а значения параметра N_p подобраны так, чтобы при его удвоении значения err (либо err_{loc}) менялись не более, чем на 1%. Для некоторых численных решений значения погрешности достигают максимума только в окрестности $x = 0$. В таком случае характеризовать точность будет именно err_{loc} . В табл. 1, 2 приведены значения $\max(err, err_{loc})$. Табл. 1 включает результаты восьми, а табл. 2 – семи экспериментов, эксперименты (тесты) пронумерованы. При проведении первых двух тестов в каждой табл. число ячеек K фиксировано. При реализации тестов с номерами 3 и больше, изначально задавался размер ячеек $d_k = 0.005$. В тестах №3 при $t = 0.4375$ метод установления совершал 200 итераций, ввиду чего (в соответствии с рекомендациями данными после (33)), размер ячейки уменьшался в 2 раза и решения пересчитывались. Подобное происходило во всех тестах с большими номерами и при различных t .

Построенный алгоритм позволяет получить баланс точности и скорости работы, важный при решении прикладных задач: этот баланс может быть достигнут за счёт изменения параметров N и K : их уменьшение ведёт с одной стороны к потере точности, а с другой – к уменьшению времени работы алгоритма. Кроме, того для ускорения алгоритма (конечно за счёт снижения точности) можно использовать следующие механизмы (они были использованы в тестах 5–8 из табл. 1 и 4–7 из табл. 2):

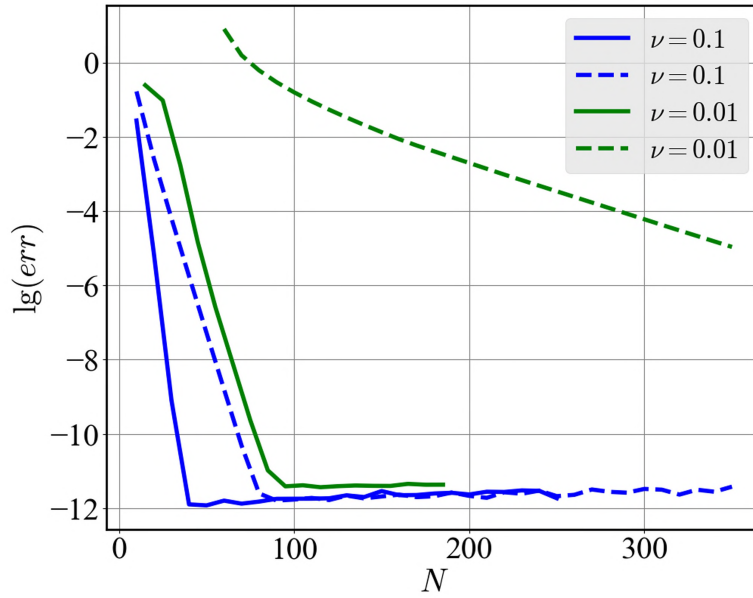


Рис. 5. Графики зависимости $\ln(err)$ от N при $T = 0.5$, $\nu = 0.1, 0.01$ в случае применения адаптивного алгоритма (сплошные линии) и алгоритма без адаптации (пунктирные линии). Параметры алгоритма: $M = 10, K = 10, H = 300$ при $\nu = 0.1$; $M = 10, K = 20, H = 300$ при $\nu = 0.01$.

- 1) Если $10^{-6} \leq \nu < 10^{-4}$, то принудительно уменьшать d_k до 0.001 после $t = 0.3$; производить поиск особенности, сгущение сетки и спектральное разложение $B_{(2)}^T$ не более одного раза на отрезке по времени длины 0.001;
- 2) Если $\nu < 10^{-6}$, то принудительно уменьшать d_k до 0.0001 после $t = 0.3$; производить сгущение сетки и спектральное разложение $B_{(2)}^T$ не более одного раза на отрезке по времени длины 0.0001.

Принудительное уменьшение шага сетки по времени, d_k , связано с тем, что в момент времени $t \approx 0.4$, мнимая часть координаты особой точки принимает минимальное значение (это следует из результатов на рис. 6, б), что приводит к возникновению больших градиентов по x и t . Для приближения функций с такими градиентами по t следует либо увеличивать число узлов M либо уменьшать d_k . Из наших экспериментов следует, что второй вариант намного предпочтительнее.

Отметим также, что при больших значениях N спектральное разложение $B_{(2)}^T$ в каждой ячейке занимает достаточно много времени, поэтому для ускорения алгоритма предлагается пропускать эту процедуру в малой окрестности момента времени, когда уже была обнаружена особая точка и сетка адаптирована. Данное предложение, естественно, ведет

к потере точности, но дает возможность ускорить работу алгоритма в несколько раз. Это можно наблюдать на примерах тестов 4–5 из табл. 1. Так в тесте 4 не была использована ни одна из двух описанных рекомендаций, что позволило получить повышенную точность решения, но время расчётов оказалось примерно в 5 раз больше, чем в тесте 5, где были применены обе рекомендации.

Также отметим, что использование композиции $g \circ g$ при $\nu \geq 10^{-5}$ позволило получить численное решение с меньшей погрешностью, чем при использовании исходного отображения g , см. (35). А при $\nu < 10^{-5}$ применение отображения $g \circ g$ оказалось единственным способом, позволившим в принципе добиться сходимости алгоритма.

Из результатов, приведённых в табл. 1, 2 можно получить несколько закономерностей. Во-первых, при достаточно больших t , $\varepsilon(t) \approx C_t \nu$, где C_t медленно растёт с ростом t . Во-вторых, в случае применения адаптивного дробления шага d_k число ячеек K приблизительно обратно пропорционально ν . В-третьих, расчёты на отрезке $t \in [0.5, 1]$ как правило требуют существенно больше времени, чем расчёты на отрезке $t \in [0, 0.5]$. Это естественно объясняется тем, что при всех $t \in [0.5, 1]$ особая точка лежит близко к области решения задачи, см. рис. 6. При этом, если использованы 2 метода ускорения алгоритма, описанные выше, погрешность решения при $t = 0.5$ приблизительно на порядок меньше, чем при $t = 1$.

Таблица 1. Погрешность численного решения задачи (18) и время расчётов в секундах *time* в зависимости от значений параметров метода при $T = 0.5$.

	ν	N	K	H	$\max(err, err_{loc})$	<i>time</i>	$\varepsilon(T)$
1	1.e-2	100	10	3e+02	1.82e-12	14с	3.37e-02
2	1.e-3	150	20	1e+03	4.57e-12	1м 30с	3.15e-03
3	1.e-4	150	125	5e+03	1.17e-10	9м 35с	3.14e-04
4	1.e-5	300	693	2e+04	4.89e-09	6ч 29м	3.14e-05
5	1.e-5	250	638	2e+04	5.73e-07	1ч 20м	3.14e-05
6	1.e-6	200	5936	2.5e+04	3.53e-07	4ч 50м	3.14e-06
7	1.e-7	250	48988	5e+04	9.18e-05	27ч	3.14e-07
8	1.e-8	250	450441	1e+05	6.83e-03	3д	3.44e-08

На рис. 6 изображена траектория движения особой точки, полученная с помощью численного алгоритма, описанного выше и использующего композицию отображений $g \circ g$. Из рисунка видно, что мнимая координата особой точки изначально стремительно приближается к своему минимальному значению до момента времени $t^* \approx 0.4$ (это значение разнится в зависимости от ν), а далее “отскакивает” от этого значения, но порядок значения $\varepsilon(t)$ при $t > t^*$ сохраняется. При малых t наблюдаются колебания действительной части координаты особой точки $\delta(t)$, данный

ТАБЛИЦА 2. Погрешность численного решения задачи (18) и время расчётов в секундах $time$ в зависимости от значений параметров метода при $T = 1$.

	ν	N	K	H	$\max(err, err_{loc})$	$time$	$\varepsilon(T)$
1	1.e-2	100	10	3e+02	1.77e-12	16с	4.31e-02
2	1.e-3	150	40	1e+03	8.66e-12	3м 02с	4.27e-03
3	1.e-4	150	325	5e+03	2.57e-10	24м 54с	4.26e-04
4	1.e-5	250	2658	2e+04	5.43e-07	3ч 44м	4.26e-05
5	1.e-6	200	25936	2.5e+04	3.73e-06	21ч	4.26e-06
6	1.e-7	250	208988	5e+04	2.93e-04	2д	4.26e-07
7	1.e-8	250	1730442	1e+05	2.02e-02	14д	4.33e-08

эффект, вероятно, возникает из-за численной погрешности метода приближения, он не так важен ввиду пренебрежимо малой амплитуды этих колебаний $\sim 10^{-7}$, рис. 6, а. Отметим также, что динамика $\varepsilon(t)$, изображённая на рис. 6, б, качественно соответствует данным приведённым в [24] при $\nu = 0.05$. Такая динамика является характерной для решения УБ при малых ν с начальными данными, представляющими целые функции: в начальный момент времени особые точки решения выходят из бесконечности, стремительно приближаются к области задачи, а затем начинают удаляться (больше информации по этому поводу приведено в [23]).

4.3. Сравнение полученных результатов с решениями других авторов. Задача (18) рассматривалась многими исследователями, см. [24, 16, 22] и ссылки, приведённые в этих работах. Однако в силу проблем с расчётом точного решения (см. раздел. 2), провести сравнение численных результатов с точным решением не удавалось. Из численных данных для сравнения в литературе присутствуют значения производной $\partial u/\partial x$ при $x = 0$ и умеренных значениях ν в различные моменты времени. Как показывают наши вычисления, и эти значения были рассчитаны с достаточно большой погрешностью.

В табл. 3 представлены значения производных решения по x в точке $x = 0$ в момент времени $T = 0.5$ при различных ν . Точное значение производной вычислено путём дифференцирования решения (11), применения поправочного множителя с функцией $\psi(t, x)$ и квадратурной формулы двойной экспоненты при $H = 10^5$ и $2H$ для контроля погрешности. Обозначим это значение

$$D_{ex}^{H,T} = \left. \frac{\partial u_{ex}}{\partial x} \right|_{x=0} \text{ при заданных } H \text{ и } T.$$

Во второй колонке табл. 3 в круглых скобках указаны 13-й – 16-й знаки значений $D_{ex}^{2H,T}$. Из этого видно, что значения производных рассчитаны с точностью как минимум 13 знаков.

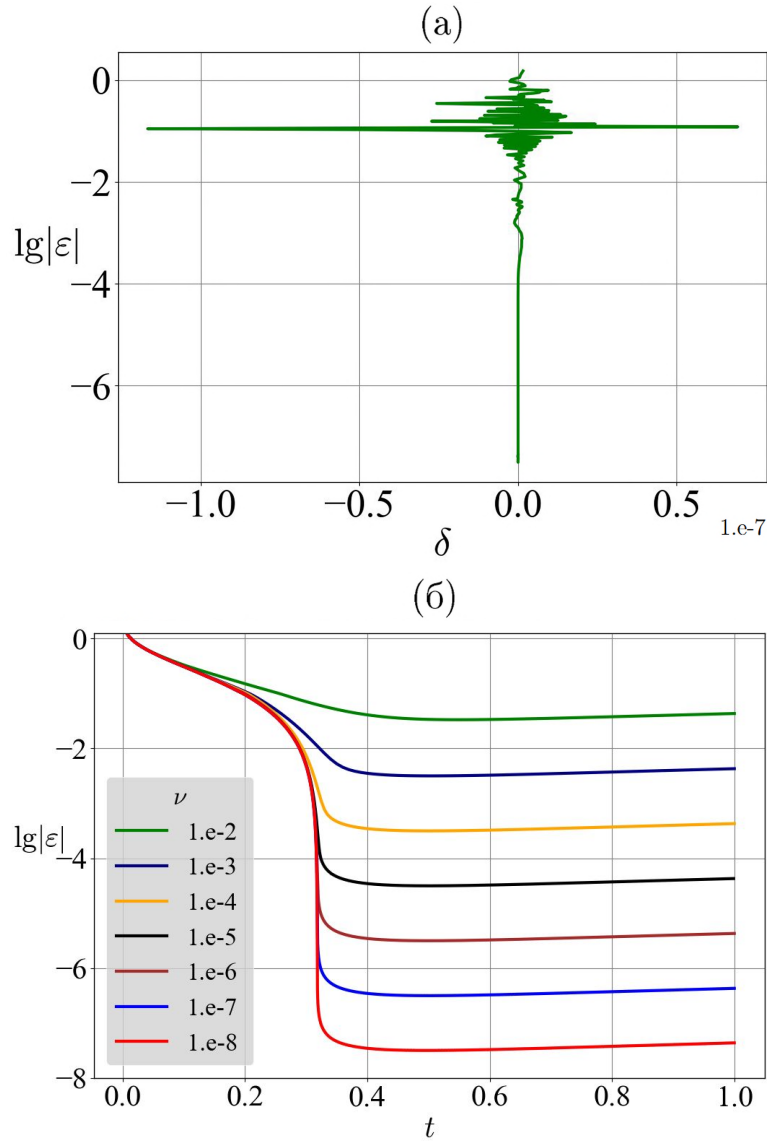


Рис. 6. Траектория движения особой в задаче (18), $t \in [0, 1]$ при $\nu = 10^{-8}$, $N = 250$, $M = 10$ и $K = 1730442$ (а); траектории движения мнимой части координат особой точки при различных ν (б).

В третьей и четвёртой колонках табл. 3 приведены приближённые значения производных, полученные путём численного дифференцирования решений, соответствующим лучшим результатам по значению погрешности из табл. 1. При этом реализованы два подхода: в первом используется барицентрическая формула без модификации (19), (23), значение

производной в точке $x = 0$ при этом обозначим D_{WM} . Во втором же подходе используются модифицированные формулы (20) и (24). Значения производных, полученные при этом, обозначим D_M .

Из табл. 3 видно, что при умеренных ν ($\nu > 10^{-5}$) значение D_M является менее точным, чем D_{WM} . Это связано с тем, что применение модификации (20) несколько ухудшает численную устойчивость метода приближения (без доказательства постулируем, что константы Лебега приближения (20) с ростом N растут быстрее, чем (19)). Тем не менее, в рамках разработанного метода выполнение модификации является необходимой процедурой, см. замечание 3. Таким образом, для поиска численного решения УБ нужно использовать формулы с модификацией, а для вычисления производных решения – исходные барицентрические формулы.

При малых ν ($\nu \leq 10^{-5}$) основной вклад в погрешности значений D_M и D_{WM} вносят погрешности численного решения, полученного с применением предложенного алгоритма, поэтому значения D_M и D_{WM} совпадают.

Таблица 3. Значения производных решения (18) по x в точке $x = 0$ при $T = 0.5$ и различных ν , точное значение производной рассчитано при $H = 10^5$ и $2H$.

ν	$-D_{ex}^{H,T}$ (4 цифры $-D_{ex}^{2H,T}$)	$-D_{WM}$	$-D_M$
1.e-2	43.88646098024938(4862)	43.8864609801	43.9
1.e-3	494.9830870739711(9639)	494.9830870732	494.982
1.e-4	4995.057436483183(3162)	4995.057435	4995.0573
1.e-5	49995.06442574773(4775)	49995.0643	49995.0643
1.e-6	499995.0651206324(6331)	499994.7	499994.7
1.e-7	4999995.065190129(0075)	5000686.2	5000686.2
1.e-8	49999995.06519861(963*)	49457792.1	49457792.1

В табл. 4 показано сравнение значений производной решения задачи (18) в нуле с результатами из [22]. В этой работе представлен алгоритм "Derivative Collocation Method using Mexican hat wavelet", а также произведено сравнение с результатами из работы [16], в которой в свою очередь обсуждаются несколько подходов к численному решению задачи (18). Отметим, что разработанный алгоритм позволяет получать решения с существенно бóльшей точностью. При этом мы используем более скрупулёзный метод оценки погрешности решений на основе расчёта err и err_{loc} . Действительно, в большинстве работ отклонения точного и приближённого решений рассчитываются в узлах сетки численного метода (отдельная более мелкая сетка не вводится), кроме того окрестности особой точки, где численные решения могут обладать высокочастотными локальными осцилляциями при расчёте погрешности исключаются, см.

[22]. Определим относительную погрешность расчёта производной решения в нуле формулой

$$err_x^T = \frac{|D_{ex}^{10^4, T} - D_{WM}|}{|D_{ex}^{10^4, T}|}.$$

В табл. 4 приведены наиболее точные значения производных из работы [22] и результаты наших вычислений (РНВ) для тех же значений t , а также указаны значения относительной погрешности. Очевидно кардинальное преимущество предложенного алгоритма.

ТАБЛИЦА 4. Сравнение результатов работы алгоритма с результатами из [22] при $\nu = 10^{-2}/\pi$, $N = 150$, $M = 10$, $K = 20$.

πT	$-D_{ex}^{10^4, T}$	$-\frac{\partial u}{\partial x} _{x=0}$ [22]	err_x^T [22]	$-D_{WM}$, РНВ	err_x^T , РНВ
1.6030	152.00508883277	149.59	1.59e-02	152.0050888329	8.49e-13
1.6035	152.00515616723	151.48	3.45e-03	152.0051561677	3.07e-12

5 Обсуждение результатов и выводы

В завершение работы прокомментируем основные качества предложенного метода. Метод позволяет проводить расчёты в задачах для уравнения Бюргерса (УБ) с предельно малыми значениями вязкости. Для рассмотренной задачи провести расчёт решений в системах с плавающей точкой при $\nu < 10^{-5}$ вплоть до значений $\nu = 10^{-8}$ и систематически исследовать погрешность, насколько нам известно, ранее никому не удавалось (по крайней мере, результаты таких исследований в литературе отсутствуют).

В этом контексте нужно отметить работу [45], где решение задачи для УБ (1) с начальным данным $u(0, x) = \varphi(x) = \cos(\pi x/2)$ и условиями $u(t, 0) = 1$, $u(t, 2) = -1$ при $x \in [0, 2]$, $t > 0$ разыскивалось с применением метода коллокаций и наименьших квадратов при уменьшении значений ν вплоть до $\nu = 10^{-14}$, $\nu = 0$. Полученные решения не имели осцилляций и сходились при $\nu \rightarrow 0$ к соответствующему решению, что подтвердило идею Хопфа [4]. Однако сравнение с точными решениями указанной задачи и исследование сходимости численных решений при фиксированных ν в [45] отсутствуют.

Ключевыми аспектами метода, предложенного в настоящей работе, с точки зрения вычисления решений при малых ν является применение дробно-рациональных аппроксимаций, учёт положения того полюса аналитического продолжения решения, который лежит ближе всего к области решения задачи, применение нового отображения (36), позволяющего существенно увеличить область аналитичности искомой функции, и

конечно предложенный подход к построению и решению задачи линейной алгебры. В последнем аспекте важным являются малые значения чисел обусловленности матриц R_A и R_B в спектральных разложениях (30), см. [42, 43]. Подчеркнём, что несмотря на наличие точных решений УБ, (11), при малых вязкостях сама задача верификации численных решений является нетривиальной: расчёт точного решения с контролем погрешности требует модификации формул точного решения. Об написано в разд. 2.

Отметим также, что предложенный алгоритм за счёт настройки его параметров, описанной в п. 4.2, позволяет достичь нужного баланса между точностью и временем вычислений, что является принципиальным для решения многих прикладных задач.

Наконец, заметим, что разработанный метод является важным для исследования процессов формирования и эволюции особенностей решений уравнений гидродинамики. Достоверно известно, что возникновение особых точек у аналитических продолжений решений соответствующих задач характеризует целый комплекс важных практических эффектов, [26, 27, 28, 29, 30]. Кроме того, приближение особой точки к области решения задачи связан с вопросом о существовании во времени гладких решений уравнений гидродинамики, что представляет известную нерешённую проблему. Разработка новых подходов к аппроксимации и поиску соответствующих решений позволит открыть новые пути к анализу данной проблемы.

References

- [1] J.M. Burgers, *A mathematical model illustrating the theory of turbulence*, Adv. Appl. Mech., **1** (1948), 171–199.
- [2] H. Bateman, *Some recent researches on the motion of fluids*, Mon. Weather Rev., **43** (1915), 163–170.
- [3] E.W. Benton, G.W. Platzman, *A table of solutions of the one-dimensional Burgers equation*, Quart. Appl. Math., **30:2** (1972), 195–212.
- [4] E. Hopf, *The partial differential equation $u_t + uu_x = \mu u_{xx}$* , Comm. Pure Appl. Math., **3** (1950), 201–230.
- [5] J.D. Cole, *On a quasi-linear parabolic equation occurring in aerodynamics*, Quart. Appl. Math., **9:3** (1951), 225–236.
- [6] J. Crank, P. Nicholson, *A practical method for numerical evaluation of solutions of partial differential equations of the heat-conduction type*, Proc. Cambridge Philos. Soc., **43** (1947), 50–67.
- [7] P.D. Lax, B. Wendroff, *Difference schemes for hyperbolic equations with high order of accuracy*, Comm. Pure Appl. Math., **17** (1964), 381–398.
- [8] I.A. Hassaniien, A.A. Salama, H.A. Hosham, *Fourth-order finite difference method for solving Burgers' equation*, Appl. Math. Comput., **170** (2005), 781–800.
- [9] T. Zhanlav, O. Chuluunbaatar, V. Ulziibayar, *Higher-order accurate numerical solution of unsteady Burgers equation*, Appl. Math. Comput., **250** (2015), 701–707.
- [10] A.H.A. Ali, G.A. Gardner, L.R.T. Gardner, *A collocation solution for Burgers equation using cubic B-spline finite elements*, Comput. Methods Appl. Mech. Eng., **100** (1992), 325–337.

- [11] T. El Sayed, A. El Danaf, *Numerical solution of the Korteweg–de Vries Burgers equation by using quintic spline method*, Stud. Univ. “Babeş-Bolyai” Math., **47**:2 (2002), 41–54.
- [12] S. Kutulay, A. Esen, I. Dag, *Numerical solutions of the Burgers’ equation by the least-squares quadratic B-spline finite element method*, J. Comput. Appl. Math., **167** (2004), 21–33.
- [13] A. Dogan, *A Galerkin finite element approach to Burgers’ equation*, Appl. Math. Comput., **157** (2004), 331–346.
- [14] Y. Ucar, M. Yagmurlu, I. Celikkaya, *Numerical solution of Burger’s type equation using finite element collocation method with Strang splitting*, Math. Sci. Appl. E-Notes, **8**:1 (2020), 29–45.
- [15] B. De Maerschalck, M.I. Gerritsma, *Least-squares spectral element method for non-linear differential equations*, J. Comput. Appl. Math., **215** (2008), 357–367.
- [16] C. Basdevant, M. Deville, P. Haldenwang, J.M. Lacroix, J. Ouazzani, R. Peyret, P. Orlandi, A.T. Patera, *Spectral and finite difference solutions of the Burgers equation*, Computers & Fluids, **14**:1 (1986), 23–41.
- [17] S. Kanoksirirath, *An integral-like numerical approach for solving Burgers’ equation*, Pure Appl. Math. J., **3**:2 (2024), 17–28.
- [18] K. Altıparmak, T. Ozis, *Numerical solution of Burgers’ equation with factorized diagonal Pade approximation*, Int. J. Numer. Methods Heat Fluid Flow, **21**:3 (2011), 310–319.
- [19] A.H. Khater, R.S. Temsah, M.M. Hassan, *A Chebyshev spectral collocation method for solving Burgers’ type equations*, J. Comput. Appl. Math., **222** (2008), 333–350.
- [20] W. Adel, H. Rezazadeh, M. Inc, *On numerical solutions of Telegraph, viscous, and modified Burgers equations via Bernoulli collocation method*, Scientia Iranica, **31**:1 (2024), 43–54.
- [21] N. Arar, B. Deghdough, S. Dekkiche, Z. Torch, A.M. Nagy, *Numerical solution of the Burgers’ equation using Chebyshev polynomials*, Int. J. Appl. Comput. Math., **10** (2024), Art. 33.
- [22] O.V. Vasilyev, S. Paolucci, M. Sen, *A multilevel wavelet collocation method for solving partial differential equations in a finite domain*, J. Comput. Phys., **120** (1995), 33–47.
- [23] D.J. VandenHeuvel, Ch.J. Lustrri, J.R. King, I.W. Turner, S.W. McCue, *Burgers’ equation in the complex plain*, Physica D, **448** (2023), Art. 133686.
- [24] C. Sulem, P.-L. Sulem, U. Frisch, *Tracing complex singularities with spectral methods*, J. Comput. Phys., **50** (1983), 138–161.
- [25] A.M. Lipanov, S.A. Karskanov, *Solution of Burgers equations by the right-hand side method*, Chemical Physics and Mesoscopy, **24**:2 (2022), 218–227.
- [26] W. Pauls, T. Matsumoto, U. Frisch, J. Bec, *Nature of complex singularities for the 2D Euler equation*, Physica D, **219** (2006), 40–59.
- [27] R.E. Caffisch, F. Gargano, M. Sammartino, V. Sciacca, *Complex singularities and PDEs*, Riv. Math. Univ. Parma, **6**:1 (2015), 69–133.
- [28] J.A.C. Weideman, *Computing the dynamics of complex singularities of nonlinear PDEs*, SIAM J. Appl. Dyn. Syst., **2**:2 (2003), 171–186.
- [29] J.A.C. Weideman, *Dynamics of complex singularities of nonlinear PDEs*, In: Recent Adv. Indust. Appl. Math., SEMA SIMAI Springer Ser., **1** (2022), 227–247.
- [30] E.A. Karabut, E.N. Zhuravleva, N.M. Zubarev, *Application of transport equations for constructing exact solutions for the problem of motion of a fluid with a free boundary*, J. Fluid Mech., **890** (2020), Art. A13.
- [31] S.N. Bernstein, *Sur l’ordre de la meilleure approximation des fonctions continues par des polynomes de degre donne*, Mem. Acad. Roy. Belg., 1912, 1–104.
- [32] L.N. Trefethen, *Approximation Theory and Approximation Practice*, SIAM, Philadelphia, 2013.

- [33] L.N. Trefethen, A. Gopal, *Solving Laplace problems with corner singularities via rational functions*, SIAM J. Numer. Anal., **57**:5 (2019), 2074–2094.
- [34] V.P. Shapeev, *Solution of the Cauchy problem for ordinary differential equations using the collocation and least squares method with the Pade approximation*, Bulletin SUSU MMCS, **16**:4 (2023), 71–83.
- [35] C. Schneider, W. Werner, *Some new aspects of rational interpolation*, Math. Comp., **47**:175 (1986), 285–299.
- [36] R. Baltensperger, J.-P. Berrut, B. Noel, *Exponential convergence of a linear rational interpolant between transformed Chebyshev points*, Math. Comp., **68** (1999), 1109–1120.
- [37] T.W. Tee, L.N. Trefethen, *A rational spectral collocation method with adaptively transformed Chebyshev grid points*, SIAM J. Sci. Comput., **28** (2006), 1798–1811.
- [38] S.V. Idimeshev, *Fractional-rational approximation in initial-boundary value problems with fronts*, Computational Technologies, **25**:2 (2020), 63–79.
- [39] L.S. Bryndin, B.V. Semisalov, V.A. Belyaev, V.P. Shapeev, *Numerical analysis of the destruction of a one-dimensional polymer fluid flow with a front*, Comput. Math. Math. Phys., **64**:1 (2024), 162–175.
- [40] B.V. Semisalov, *Application of fractional-rational interpolations for solving boundary value problems with singularities*, Bulletin SUSU MMCS, **15**:4 (2022), 5–19.
- [41] B.V. Semisalov, *A fast nonlocal algorithm for solving Neumann–Dirichlet boundary value problems with error control*, Numer. Methods Program., **17**:4 (2016), 500–522.
- [42] B.V. Semisalov, S.B. Medvedev, M.P. Fedoruk, S.V. Nazarenko, *Algorithm for solving the four-wave kinetic equation in problems of wave turbulence*, Comput. Math. Math. Phys., **64**:2 (2024), 364–386.
- [43] B.V. Semisalov, *Modeling of flows of viscoelastic polymeric media and weakly turbulent processes in Bose gases based on fractional-rational approximations and unsaturable algorithms*, Diss. ... Dr. Phys.-Math. Sci.
- [44] H. Takahasi, M. Mori, *Double exponential formulas for numerical integration*, Publ. Res. Inst. Math. Sci., **9**:3 (1973), 721–741.
- [45] V.P. Shapeev, E.V. Vorozhtsov, *CAS application to the construction of the collocations and least residuals method for the solution of the Burgers and Korteweg–de Vries–Burgers equations*, In: Computer Algebra in Scientific Computing (CASC 2014), Lecture Notes in Comput. Sci., **8660** (2014), 432–446.

IVAN ANDREEVICH BUGOETS

SOBOLEV INSTITUTE OF MATHEMATICS; NOVOSIBIRSK STATE UNIVERSITY,
KOPTYUGA AVENUE, 4; PIROGOVA STREET, 1,
630090, NOVOSIBIRSK, RUSSIA
Email address: i.bugoets@ng.su.ru

BORIS VLADIMIROVICH SEMISALOV

SOBOLEV INSTITUTE OF MATHEMATICS; FEDERAL RESEARCH CENTER FOR INFORMATION
AND COMPUTATIONAL TECHNOLOGIES; NOVOSIBIRSK STATE UNIVERSITY,
KOPTYUGA AVENUE, 4; LAVRENTYEV AVENUE, 6; PIROGOVA STREET, 1,
630090, NOVOSIBIRSK, RUSSIA
Email address: vibis87@gmail.com

VASILY PAVLOVICH SHAPEEV

NOVOSIBIRSK STATE UNIVERSITY,
PIROGOVA STREET, 1,
630090, NOVOSIBIRSK, RUSSIA
Email address: shapeev.vasily@mail.ru