

УДК 519.622.2

# БИБЛИОТЕКА ПРОГРАММ CCODE ДЛЯ РЕШЕНИЯ ЖЁСТКИХ СИСТЕМ ОБЫКНОВЕННЫХ ДИФФЕРЕНЦИАЛЬНЫХ УРАВНЕНИЙ

© 2025 К. П. Сактоева<sup>a</sup>, Ю. М. Лаевский<sup>b</sup>

*Институт вычислительной математики  
и математической геофизики СО РАН,  
просп. Акад. Лаврентьева, 6, г. Новосибирск 630090, Россия*

E-mails: <sup>a</sup>saktueva33@gmail.com, <sup>b</sup>laev@labchem.sccc.ru

Поступила в редакцию 24.05.2025 г.; после доработки 17.06.2025 г.;  
принята к публикации 17.09.2025 г.

Рассмотрена библиотека программ для решения жёстких систем обыкновенных дифференциальных уравнений, основанная на  $(m, k)$ -методах переменного шага интегрирования и протестированная на некоторых системах, описывающих процессы химической кинетики. Основное внимание уделено выбору наиболее эффективных схем и сравнению их производительности с широко используемой программой RADAU5.

**Ключевые слова:** библиотека программ, жёсткие системы, одношаговые методы, химическая кинетика, численный эксперимент.

DOI: 10.33048/SIBJIM.2025.28.306

## Введение

В статье дано краткое описание разработанной в Институте вычислительной математики и математической геофизики СО РАН библиотеки программ CCODE для решения жёстких систем обыкновенных дифференциальных уравнений (ОДУ). Речь идёт о задаче Коши для автономной системы уравнений:

$$\begin{aligned}\frac{dy}{dt} &= f(y), \quad t > 0, \\ y(0) &= y_0,\end{aligned}$$

где  $y$ ,  $f$  и  $y_0$  — векторы-столбцы размерности  $d$ . Вектор  $y(t)$  — искомое решение,  $y_0$  — заданный вектор. Тот факт, что рассматривается автономная система, не ограничивает общности, поскольку всегда можно расширить систему, вводя дополнительную компоненту решения  $y_{d+1} = t$ . Краткость описания означает, что данная работа не может заменить программную документацию с детальным рассмотрением всех функций библиотеки, включая входную/выходную информацию. Основное внимание в работе уделено выбору методов интегрирования и соответствующих программных кодов из заданного класса для решения некоторых практических задач. Понятие жёсткости системы ОДУ и связанные с ним проблемы построения методов интегрирования хорошо известны (см., например, монографию [1]) и здесь не обсуждается. Важным поставщиком жёстких систем являются процессы химической кинетики, и именно решению такого сорта задач уделено основное внимание при разработке библиотеки CCODE. Отметим, что одной из основных мотиваций данной работы явилась потребность в такой библиотеке для решения задач химической кинетики в пакете программ «ЛОГОС» [2].

Методической основой библиотеки CCODE является класс одношаговых безытерационных  $(m, k)$ -методов, предложенных Е.А. Новиковым в 1988 году [3]. Эти методы являются

обобщением схемы Розенброка, существенно повышающим их производительность [4]. В пункте 1 статьи будут рассмотрены некоторые конкретные  $(m, 2)$ -методы, детально изученные в статье [5], и составляющие алгоритмическую базу библиотеки. А именно, будут приведены  $L$ -устойчивые  $(4, 2)$ -методы 3-го порядка точности [6] и  $(5, 2)$ -методы 3-го и 4-го порядков точности [7] с различными наборами параметров, задающими конкретные методы. При этом важным элементом алгоритмов, фактически определяющим их эффективность, является стратегия изменения величины шага интегрирования, обеспечивающая заданную точность вычислений. Наиболее эффективной, на наш взгляд, стратегией является вычисление глобальной погрешности на основе вложенных методов (см., например, [1]). Для  $(m, k)$ -методов, соответствующие результаты приведены в [8]. В отличие от общепринятого подхода, когда для оценки погрешности дополнительно к основному методу строится метод более высокого порядка, Е.А. Новиковым было предложено использовать метод более низкого порядка, использующий вычисленные стадии основного метода. При этом удалось сконструировать алгоритмы, в которых наряду с основной схемой промежуточные схемы также являются  $L$ -устойчивыми. Именно такие методы приведены в статьях [6], [7]. Изложение пункта 1 данной статьи в целом соответствует работе [9].

Структура библиотеки CCODE кратко описана в пункте 2. Здесь перечислены функциональные модули библиотеки без детального описания соответствующих интерфейсов. Отметим, что в начале 2000-х годов Е.А. Новиков передал одному из соавторов настоящей статьи комплекс ФОРТРАН-программ, реализующий некоторую стратегию перехода с явного метода на неявный и обратно в зависимости от текущей жёсткости системы ОДУ. При этом в качестве неявного метода в комплексе использовался  $(5, 2)$ -метод 3-го порядка точности. Такого сорта универсальный подход зачастую приводит к большим накладным расходам, связанными с анализом жёсткости задачи, на основании которого программа принимает решение о переключении на другой метод. Предназначение библиотеки CCODE состоит не в решении систем с заранее неизвестной жёсткостью, а в её использовании для решения задач химической кинетики, являющихся, как правило, чрезвычайно жёсткими. Именно поэтому библиотека включает набор C++ программ, реализующих методы из пункта 1, и в настоящей версии библиотеки конкретный выбор программы предоставлен пользователю. Собственно, одной из целей данной статьи является обеспечение пользователя определёнными рекомендациями относительно выбора метода. Библиотека содержит  $(4, 2)$ -метод 3-го порядка точности с двумя наборами параметров и  $(5, 2)$ -метод 4-го порядка с четырьмя наборами. Выбору лучших наборов параметров посвящён пункт 3, в котором проводится экспериментальное сравнение производительности методов на реакции Робертсона, являющейся популярным модельным примером из области химической кинетики [10].

Далее, в пункте 4 проводится тестирование CCODE на трёх задачах из области химической кинетики. Это лазерный пиролиз этана [11], процессы горения водорода [12] и малых углеводородов [13]. Соответствующие задачи Коши приведены в Приложении. Для этих задач осуществлено сравнение производительности библиотеки CCODE с программой RADAU5, разработанной Э. Хайрером и Г. Ванером на основе метода Радо IIА [1] и опубликованной в виде текста ФОРТРАН-программы в этой же монографии. Мы выбрали для сравнения RADAU5, поскольку, на наш взгляд, на задачах химической кинетики эта программа является одной из самых эффективных и работает не хуже таких известных пакетов, как LSODE [14], VODE [15] и др., разработанных в Ливерморской лаборатории (LLNL) на основе метода Гира [16]. При тестировании использовалась программа из [17].

И, наконец, в пункте 5 приводятся некоторые комментарии, связанные с дополнительными возможностями библиотеки CCODE и её дальнейшим развитием.

Авторы посвящают данную работу памяти профессора Е.А. Новикова, чьи фундаментальные и прикладные результаты легли в основу разрабатываемой библиотеки программ.

### 1. Методы интегрирования и оценка погрешности

В этом пункте приводятся  $L$ -устойчивые  $(4, 2)$ - и  $(5, 2)$ -методы с  $L$ -устойчивыми внутренними схемами, реализованные в библиотеке CCODE. В дальнейшем используется обозначение

$$D_n = E - ahJ_n, \quad (1)$$

где  $J_n = \partial f / \partial y(y_n)$  – матрица Якоби системы ОДУ в точке  $y_n$ ,  $E$  – единичная матрица,  $a$  – числовой параметр,  $h$  – текущий шаг интегрирования.  $(4, 2)$ -метод с внутренней схемой записывается в виде:

$$y_{n+1} = y_n + p_1 k_1 + p_2 k_2 + p_3 k_3 + p_4 k_4, \quad (2)$$

$$D_n k_1 = hf(y_n),$$

$$D_n k_2 = k_1,$$

$$D_n k_3 = hf(\tilde{y}_{n+1}) + \alpha_{32} k_2,$$

$$D_n k_4 = k_3 + \alpha_{42} k_2,$$

где внутренняя схема задаётся равенством

$$\tilde{y}_{n+1} = y_n + \beta_{31} k_1 + \beta_{32} k_2. \quad (3)$$

$L$ -устойчивость метода с  $L$ -устойчивой внутренней схемой обеспечивают два набора параметров, приведённых в табл. 1 и используемых в библиотеке CCODE.

Таблица 1

Коэффициенты  $(4, 2)$ -метода

Набор 1	Набор 2	
$a$	1.2803300858899	0.2196699141101
$p_1$	1.2803300858899	0.2196699141101
$p_2$	-0.8138796466463	0.4126450787451
$p_3$	1.0694742839250	0.5107726296546
$p_4$	-0.4768816913329	0.0818199629379
$\beta_{31}$	1.2803300858899	0.2196699141101
$\beta_{32}$	-0.5303300858899	0.5303300858899
$\alpha_{32}$	-0.9483253348642	-9.6766746651350
$\alpha_{42}$	-1.0546169964430	67.335866996443

Рассмотрим вопрос о реализованной в библиотеке стратегии выбора шага интегрирования  $(4, 2)$ -метода 3-го порядка, основанной на оценке глобальной погрешности, которая, в свою очередь, вычисляется с использованием технологии вложенных методов. Наряду с (2) используется равенство

$$z_{n+1} = y_n + r_2 k_2 + r_3 k_3, \quad (4)$$

где  $k_2$  и  $k_3$  – стадии схемы (2), а  $r_2$  и  $r_3$  вычисляются по формулам

$$r_2 = 1 - (1 + \alpha_{32})r_3, \quad r_3 = \frac{1/2 - 2a}{3/4 - a + a\alpha_{32}}.$$

Для таких коэффициентов (4) является схемой 2-го порядка точности. Введём вектор

$$e_n = \sum_{i=1}^m \gamma_i k_i, \quad (5)$$

где  $\gamma_i = p_i - r_i$ ,  $m$  – количество стадий. В рассматриваемом случае  $m = 4$ ,  $r_1 = r_4 = 0$ . При этом, согласно (2), (4),  $e_n = y_{n+1} - z_{n+1}$ . Выбор шага  $h$  основан на контроле за выполнением неравенства

$$\varepsilon_n(j_n) = \max_{k=1,\dots,d} \frac{|(D_n^{1-j_n} e_n)_k|}{|y_{n,k}| + \rho} \leq \varepsilon, \quad j_n = 1, 2, \quad (6)$$

где индекс  $k$  означает номер компоненты вектора,  $\rho$  – положительное число, отвечающее за переход от контроля относительной ошибки  $\varepsilon$  к контролю абсолютной ошибки  $e = \rho\varepsilon$  в случае выполнения неравенства  $|y_n^i| < \rho$ . Наличие матрицы  $D_n^{1-j_n}$  связано с отсутствием  $L$ -устойчивости вспомогательной схемы (4), и при  $j_n = 1$  главный член глобальной ошибки не стремится к нулю при  $h \rightarrow \infty$ . Это негативно сказывается на производительности метода при резком увеличении шага. Ситуацию исправляет возможность использования значения  $j_n = 2$ . Далее, положим

$$\tilde{q}(j_n) = \left( \frac{\varepsilon}{\varepsilon_n(j_n)} \right)^{1/p}, \quad (7)$$

где  $p$  – порядок метода. В рассматриваемом случае  $p = 3$ . Кроме того, во избежание резкого увеличения или уменьшения шага интегрирования, что может приводить к неоправданным колебаниям шага, будем полагать:  $q(j_n) = \tilde{q}(j_n)$  при  $0.8 < \tilde{q}(j_n) < 1.2$ ,  $q(j_n) = 0.8$  в случае  $\tilde{q}(j_n) \leq 0.8$ , и  $q(j_n) = 1.2$  в случае  $\tilde{q}(j_n) \geq 1.2$ .

Пусть известны текущий шаг  $h_n$  и вектор приближённого решения  $y_n$ . Процесс вычислений на  $n$ -м шаге состоит в выполнении следующего набора действий.

Шаг 1. Вычисляются правая часть  $f(y_n)$  и матрица Якоби  $J_n$ .

Шаг 2. В соответствии с (1) формируется матрица  $D_n$  при  $h = h_n$  и осуществляется её  $LU$ -декомпозиция.

Шаг 3. В соответствии с (2), (3) последовательно вычисляются векторы  $k_1, k_2, \tilde{y}_{n+1}, f(\tilde{y}_{n+1}), k_3, k_4$ .

Шаг 4. В соответствии с формулами (5)–(7) вычисляются вектор  $e_n$  и величины  $\varepsilon_n(1), q(1)$ .

Шаг 5. Если  $q(1) \geq 1$ , переходим на шаг 7. В противном случае ( $q(1) < 1$ ) в соответствии с (6), (7) вычисляем  $\varepsilon_n(2), q(2)$ .

Шаг 6. Если  $q(2) \geq 1$ , переходим на шаг 8. В противном случае ( $q(2) < 1$ ) полагаем  $h_{new} = q(2)h_n$ , и, полагая  $h_n = h_{new}$ , возвращаемся на шаг 2.

Шаг 7. По формуле (2) вычисляем вектор  $y_{n+1}$ , и, полагая  $h_{n+1} = q(1)h_n$ , переходим на  $n + 1$ -й шаг метода.

Шаг 8. По формуле (2) вычисляем вектор  $y_{n+1}$ , и, полагая  $h_{n+1} = q(2)h_n$ , переходим на  $n + 1$ -й шаг метода.

Перейдём к рассмотрению (5, 2)-метода с одной внутренней схемой. Имеем:

$$y_{n+1} = y_n + p_1 k_1 + p_2 k_2 + p_3 k_3 + p_4 k_4 + p_5 k_5, \quad (8)$$

$$D_n k_1 = h f(y_n),$$

$$D_n k_2 = k_1,$$

$$D_n k_3 = h f(\tilde{y}_{n+1}) + \alpha_{32} k_2,$$

$$D_n k_4 = k_3 + \alpha_{42} k_2,$$

$$D_n k_5 = k_4,$$

где внутренняя схема имеет вид (3).

Т а б л и ц а 2

Коэффициенты (5, 2)-метода 4-го порядка

Набор 0	Набор 1	Набор 2	Набор 3	Набор 4
$a$	1.2803300858899	1.2803300858899	0.2196699141101	0.2196699141101
$p_1$	1.2803300858899	1.2803300858899	0.2196699141101	0.2196699141101
$p_2$	-2.9633753074324	-0.4126555970145	0.2668352254833	0.4223322710492
$p_3$	3.1291760925648	1.3255448884221	0.4018412761404	0.5117942753850
$p_4$	-4.5962853086115	-0.9890229003261	0.2996826699665	0.0797766714772
$p_5$	2.0597018086393	0.2560706044966	-0.1089313535143	0.0010216457303
$\beta_{31}$	1.2803300858899	1.2803300858899	0.2196699141101	0.2196699141101
$\beta_{32}$	-0.5303300858899	-0.5303300858899	0.5303300858899	0.5303300858899
$\alpha_{32}$	0.0435955592067	-2.5668493086922	-2.3385478649438	-10.481948385463
$\alpha_{42}$	-0.8139366291378	-1.4473367655718	6.8503244659407	73.973448927883

Процесс выбора шага интегрирования аналогичен приведённому выше для (4, 2)-метода. Вместо (4) рассмотрим следующую вспомогательную схему 3-го порядка точности:

$$z_{n+1} = y_n + r_1 k_1 + r_2 k_2 + r_3 k_3 + r_4 k_4, \quad (9)$$

где  $k_1, \dots, k_4$  – стадии схемы (8), а  $r_1, \dots, r_4$  вычисляются по следующим формулам:

$$\begin{aligned}
 r_1 &= \frac{11}{27} - r_2 - \alpha_{42} r_4 - \frac{16}{27} \alpha_{32}, \\
 r_2 &= \frac{1}{18a} - 1 - \frac{32}{27} \alpha_{32} - (1 + \alpha_{32} + 2\alpha_{42}) r_4, \\
 r_3 &= \frac{16}{27} - r_4, \\
 r_4 &= \left( \frac{43}{27} a^2 - \frac{13}{9} a + \frac{1}{6} - \frac{16}{27} a^2 \alpha_{32} \right) / \left( 2a^2 \alpha_{32} + a^2 \alpha_{42} + \frac{3}{4} a \right).
 \end{aligned}$$

Процесс вычислений совпадает с описанным выше с той лишь разницей, что вместо (2) используется равенство (8) с дополнительной пятой стадией, в (5) и в (7) полагается  $m = 5$  и  $p = 4$  соответственно, и коэффициенты  $\gamma_i$  вычисляются с учётом формул для коэффициентов  $r_i$ , входящих в (9).

## 2. Описание библиотеки CCODE

Библиотека CCODE предоставляет пользователю для решения системы ОДУ два основных комплекса программ *ccodea* и *ccoden*, отличающиеся друг от друга вычислением матрицы Якоби  $J_n$  с использованием её явного вида и численного представления на основе задаваемой правой части уравнений соответственно. В данном пункте дано описание только комплекса программ *ccodea*, для которого в пп. 3 и 4 приводятся численные результаты. Интерфейс библиотеки CCODE представлен на языке программирования C++ как набор следующих программ:

`ccodea.cpp` — предоставляет пользователю программу, реализующую один из методов решения задачи Коши с аналитическим вычислением матрицы Якоби, инициализирует выбранный пользователем набор коэффициентов;

`coef.cpp` — программа, задающая коэффициенты для определённого  $(m, 2)$ -метода в зависимости от значения входного параметра `ipar[40]`: при

$$ipar[40] = 1 / ipar[40] = 2 / ipar[40] = 3 / ipar[40] = 4$$

используется набор 1/набор 2/набор 3/набор 4 коэффициентов (5,2)-метода четвёртого порядка, при  $ipar[40]=5/ipar[40]=6$ , используется набор 1/набор 2 коэффициентов (4,2)-метода третьего порядка;

`ccode42.cpp` — реализует пошаговый алгоритм (см. п.1) для (4,2)-метода с обращением к программам пользователя, задающим вычисление правой части и матрицы Якоби системы ОДУ, входящих в список формальных параметров: *rhs* и *jacmat* соответственно;

`ccode52.cpp` — аналогично `ccode42.cpp` реализует пошаговый алгоритм для (5,2)-метода 4-го порядка точности;

`des.cpp` — осуществляет *LU*-декомпозицию заданной формулой (1) матрицы  $D_n$  с выбором главного элемента по столбцу, контролируется интервал устойчивости, если появляются собственные числа матрицы Якоби с положительной вещественной частью;

`sol.cpp` — используя представление  $D_n = L_n U_n$ , находит стадии  $k_i$  решением двух систем с треугольными матрицами:  $L_n u_i = g_i$  и  $U_n k_i = u_i$ .

В качестве формальных параметров процедура *ccodea* содержит всю необходимую для решения системы ОДУ входную информацию: размерность системы  $d$ , решение  $y$  (при первом обращении к *ccodea* содержит начальные данные  $y_0$ ), конец интервала интегрирования  $t_{end}$ , *rhs* и *jacmat* — функции, задающие вектор правой части и матрицу Якоби системы соответственно, минимальный шаг интегрирования  $h_{min}$ , относительная ошибка  $\varepsilon$ , величина  $\rho$ , регулирующая переход от контроля относительной ошибки к контролю абсолютной ошибки, выбор конкретного метода по значению  $ipar[40]$ . В качестве фактических параметров, определяющих функции для вычисления вектора правой части и матрицы Якоби, пользователь может задавать любые имена. Но структура этих процедур жёстко определена. Для процедуры, задающей правую часть системы, имеем:

```
void <...> (int * d, double * y, double * f)
{
    ...
    f[i] = fi+1(y);    // i = 0, ..., d - 1
    ...
}
```

Матрица Якоби задаётся следующей процедурой:

```
void <...> (int * d, double * y, double * a)
{
    ...
    a[i + d * j] = dfi+1/dyj+1(y);    // i, j = 0, ..., d - 1
    ...
}
```

Отметим, что массив  $a[k]$  упорядочен по столбцам матрицы Якоби.

На рис. 1 схематично изображена часть библиотеки CCODE, использующая задаваемую пользователем матрицу Якоби системы ОДУ. В комплексе программ *ccoden* соответствующая процедура отсутствует, и матрица Якоби вычисляется по заданной пользователем правой части системы.

### 3. Численный выбор методов

Данный пункт посвящён экспериментальному выбору наилучших с точки зрения производительности наборов параметров для (4, 2)- и (5, 2)-методов. При этом выборе мы не используем традиционный подход, предполагающий тестирование методов на достаточно большом

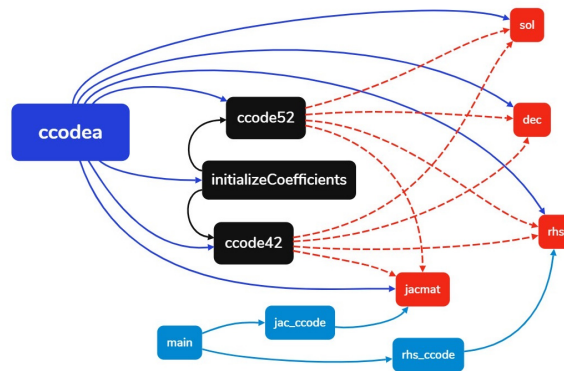


Рис. 1. Схема функционирования библиотеки CCODE

наборе задач, а проводим расчёты на хорошо известной модельной задаче Робертсона из области химической кинетики, предложенной в статье [10] (см. также [1], стр. 11 в русскоязычном издании). На интервале  $(0, t_{\text{end}}]$ ,  $t_{\text{end}} = 10^{11}$  с рассматривается система

$$\begin{aligned} \frac{dy_1}{dt} &= -0.04y_1 + 10^4 y_2 y_3, \\ \frac{dy_2}{dt} &= 0.04y_1 - 10^4 y_2 y_3 - 3 \cdot 10^7 y_2^2, \\ \frac{dy_3}{dt} &= 3 \cdot 10^7 y_2^2 \end{aligned} \quad (10)$$

с начальными данными

$$y_1(0) = 1, \quad y_2(0) = y_3(0) = 0.$$

В [17] приведено точное (до 16 значащих цифр) решение задачи на момент времени  $t_{\text{end}}$ :

$$\begin{aligned} y_1(t_{\text{end}}) &= 0.2083340149701284 \cdot 10^{-7}, \\ y_2(t_{\text{end}}) &= 0.8333360770334744 \cdot 10^{-13}, \\ y_3(t_{\text{end}}) &= 0.9999999791665152. \end{aligned} \quad (11)$$

Обычно в качестве оценки производительности метода используют количества вычислений правой части и матрицы Якоби, а также количество обратных ходов метода Гаусса. Этот, в целом объективный подход для оценки производительности, требует наличия информации о затратах времени на указанные вычисления, которые самым существенным образом зависят от решаемой задачи – её размерности, заполненности матрицы Якоби и пр. В связи с этим производительность при выборе предпочтительного набора коэффициентов оценивается непосредственными замерах времени расчёта, как среднее время из 10 запусков задачи Робертсона. Отметим, что по окончании решения задачи в массиве *ipar* содержится информация об указанных выше величинах. Время измеряется C++ функцией *steady\_clock*, использование которой может быть представлено в виде:

```

...
autotime_begin = std::chrono::steady_clock::now();
...
autotime_end = std::chrono::steady_clock::now();
...

```

Все, представленные в этом и следующем пункте результаты расчётов, получены на компьютере с процессором 12th Gen Intel® Core™ i5-12450H 2.00 GHz под управлением ОС Windows 11.

В Таблице 3 приведены времена решения задачи Робертсона (4,2)-методом при двух вариантах коэффициентов, представленных в Таблице 1. При этом использовались значения  $\rho = 1$  ( $\rho$  – параметр, определяющий абсолютную погрешность через задаваемую пользователем относительную погрешность  $\varepsilon$ ) и  $h_0 = 10^{-3}$ с (задаваемый пользователем начальный шаг интегрирования). Для наглядности на рисунке 2 приведены соответствующие диаграммы.

Таблица 3

Время решения (4,2)-методом,  $\rho = 1$ ,  $h_0 = 10^{-3}$ с

$\varepsilon$	$10^{-7}$	$10^{-6}$	$10^{-5}$	$10^{-4}$	$10^{-3}$	$10^{-2}$
Набор 1	$3.05 \cdot 10^{-3}$	$1.66 \cdot 10^{-3}$	$7.10 \cdot 10^{-4}$	$3.79 \cdot 10^{-4}$	$2.79 \cdot 10^{-4}$	$2.94 \cdot 10^{-4}$
Набор 2	$1.10 \cdot 10^{-3}$	$5.96 \cdot 10^{-4}$	$3.94 \cdot 10^{-4}$	$2.77 \cdot 10^{-4}$	$2.63 \cdot 10^{-4}$	$2.68 \cdot 10^{-4}$

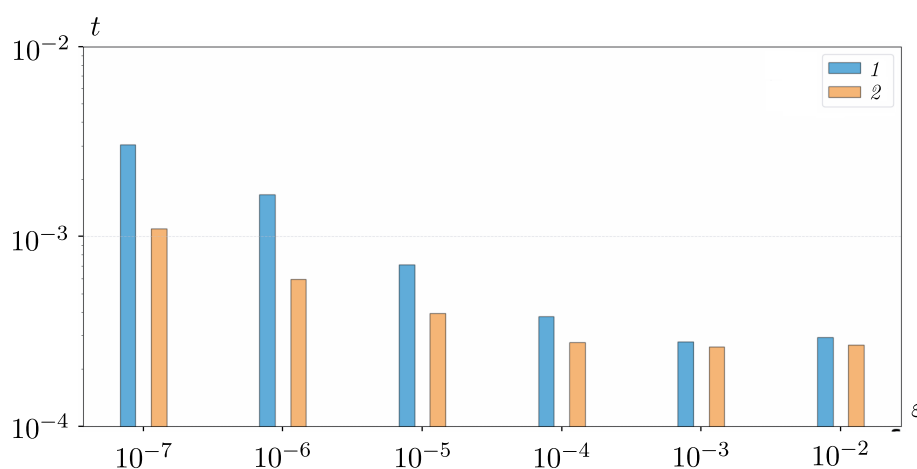


Рис. 2. Время решения (4,2)-методом,  $\rho = 1$ ,  $h_0 = 10^{-3}$  с;  
1 – набор 1; 2 – набор 2

Таким образом, с точки зрения производительности выигрывает метод со вторым набором коэффициентов.

Приведём результаты о погрешности (4,2)-метода. В Таблице 4 указаны чебышёвские нормы векторов разности приближённого решения в момент времени  $t_{end}$  и соответствующего точного решения (11). В дальнейшем эти значения мы сравним с погрешностями (5,2)-метода. Отметим, что метод со вторым набором коэффициентов даёт существенно более точные результаты, по сравнению с первым набором.

Таблица 4

Погрешность для (4,2)-метода,  $\rho = 1$ ,  $h_0 = 10^{-3}$ с

$\varepsilon$	$10^{-7}$	$10^{-6}$	$10^{-5}$	$10^{-4}$	$10^{-3}$	$10^{-2}$
Набор 1	$6.6 \cdot 10^{-10}$	$6.5 \cdot 10^{-10}$	$6.2 \cdot 10^{-10}$	$6.7 \cdot 10^{-10}$	$6.4 \cdot 10^{-10}$	$6.0 \cdot 10^{-10}$
Набор 2	$1.5 \cdot 10^{-12}$	$1.5 \cdot 10^{-12}$	$1.4 \cdot 10^{-12}$	$1.4 \cdot 10^{-12}$	$1.4 \cdot 10^{-12}$	$1.5 \cdot 10^{-12}$

Решение вопроса о выборе параметра  $\rho$  зависит от конкретной задачи и предоставлено пользователю. В данном описании работы библиотеки CCODE при выборе этого параметра мы ориентируемся на порядок метода. Так для (4,2)-метода 3-го порядка точности вычисления проводятся при  $\rho = 1$ , а для (5,2)-метода 4-го порядка будет использовано значение  $\rho = 10^{-3}$ . Отметим, что для метода 5-го порядка RADAU5 в [1] обсуждается решение задачи Робертсона при  $\rho = 10^{-6}$ . При этом расчёты показывают, что при  $\rho = 1$  RADAU5 не справляется с





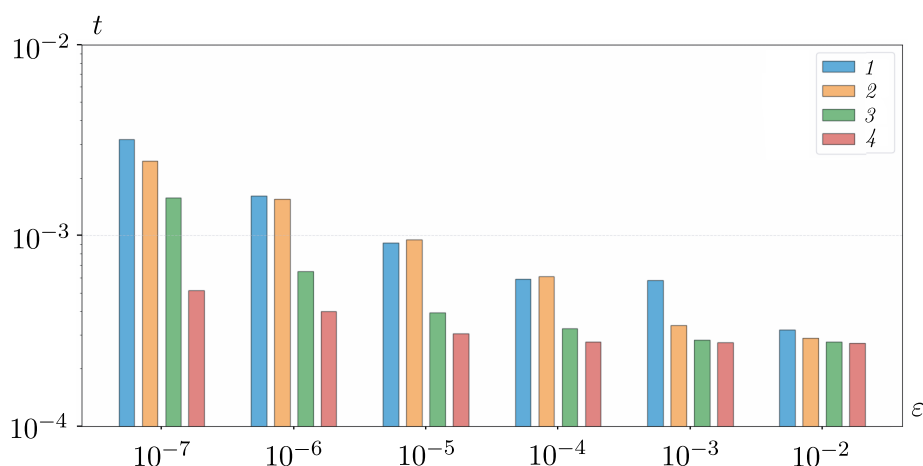


Рис. 3. Время решения (5,2)-методом,  $\rho = 10^{-3}$ ,  $h_0 = 10^{-3}$ с;  
1 — набор 1; 2 — набор 2; 3 — набор 3

Как видно из табл. 4 и 7, абсолютные ошибки решений (4,2)- и (5,2)-метода практически совпадают — сказывается специфика задачи Робертсона. При этом (5,2)-метод экономичнее (4,2)-метода при  $\varepsilon \leq 10^{-4}$ , и, начиная с  $\varepsilon = 10^{-4}$ , производительность методов совпадает и увеличивается при  $\varepsilon \geq 10^{-3}$ .

Проведём сравнение (4,2)-метода со вторым набором коэффициентов и (5,2)-метода с четвёртым набором коэффициентов с точки зрения времени решения задачи. Кроме того, для сравнения приведём результаты, полученные методом RADAU5 при  $\rho = 10^{-6}$ . Напомним, что нами использовалась программа RADAU5 из [17]. При этом, чтобы поставить все методы в равные условия по контролю погрешности, для (4,2)- и (5,2)-методов, как и в RADAU5, полагаем  $\rho = 10^{-6}$ . Результаты сравнения времени решения представлены в Таблице 8 и на рисунке 4.

Таблица 8

Время решения (4,2)-, (5,2)-методами и RADAU5 при  $\rho = 10^{-6}$

$\varepsilon$	$10^{-7}$	$10^{-6}$	$10^{-5}$	$10^{-4}$	$10^{-3}$	$10^{-2}$
(4,2)-метод	$1.61 \cdot 10^{-2}$	$5.64 \cdot 10^{-3}$	$2.09 \cdot 10^{-3}$	$8.50 \cdot 10^{-4}$	$3.91 \cdot 10^{-4}$	$3.16 \cdot 10^{-4}$
(5,2)-метод	$2.01 \cdot 10^{-3}$	$4.74 \cdot 10^{-4}$	$3.84 \cdot 10^{-4}$	$4.43 \cdot 10^{-4}$	$3.47 \cdot 10^{-4}$	$3.22 \cdot 10^{-4}$
RADAU5	$8.22 \cdot 10^{-3}$	$6.72 \cdot 10^{-3}$	$5.25 \cdot 10^{-3}$	$4.37 \cdot 10^{-3}$	$3.84 \cdot 10^{-3}$	$3.79 \cdot 10^{-3}$

В табл. 9 приведены погрешности для всех трёх методов.

Таблица 9

Погрешность для (4,2)-, (5,2)-методов и RADAU5 при  $\rho = 10^{-6}$

$\varepsilon$	$10^{-7}$	$10^{-6}$	$10^{-5}$	$10^{-4}$	$10^{-3}$	$10^{-2}$
(4,2)-метод	$2.4 \cdot 10^{-15}$	$7.0 \cdot 10^{-15}$	$7.6 \cdot 10^{-14}$	$7.1 \cdot 10^{-13}$	$1.4 \cdot 10^{-12}$	$1.5 \cdot 10^{-12}$
(5,2)-метод	$2.8 \cdot 10^{-13}$	$9.7 \cdot 10^{-13}$	$1.3 \cdot 10^{-12}$	$1.4 \cdot 10^{-12}$	$1.4 \cdot 10^{-12}$	$1.4 \cdot 10^{-10}$
RADAU5	$2.8 \cdot 10^{-13}$	$2.6 \cdot 10^{-12}$	$2.4 \cdot 10^{-11}$	$2.4 \cdot 10^{-10}$	$2.2 \cdot 10^{-9}$	$1.1 \cdot 10^{-8}$

При высокой задаваемой относительной погрешности (5,2)-метод и RADAU5 показывают примерно одинаковую точность. Несколько завышенную точность даёт (4,2)-метод. Таким образом, в диапазоне задаваемой относительной погрешности от  $\varepsilon = 10^{-7}$  до  $\varepsilon = 10^{-2}$  при

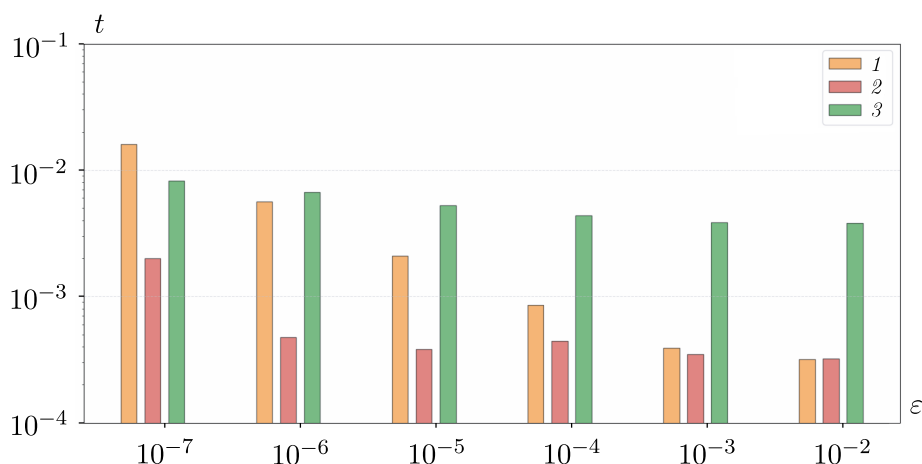


Рис. 4. Время решения (4,2)-, (5,2)-методами и RADAU5 при  $\rho = 10^{-6}$ ; 1 — набор 1; 2 — набор 2; 3 — набор 3

$\rho = 10^{-6}$  наилучшее время решения демонстрирует (5,2)-метод с четвёртым набором коэффициентов. Отметим, что при небольшой относительной погрешности  $\epsilon = 10^{-2}$  и  $\rho = 1$  (4,2)-метод немного выигрывает по времени у (5,2)-метода. При этом, как указывалось выше, RADAU5 не работает.

#### 4. Сравнение производительности с RADAU5

В данном пункте для трёх задач из области химической кинетики будут приведены результаты сравнения (4,2)-метода со вторым набором коэффициентов, (5,2)-метода с четвёртым набором коэффициентов и RADAU5. Параметры  $\rho$  и  $h_0$  будут подбираться для каждой задачи экспериментально. При этом следует учитывать, что для задач химической кинетики кроме реализуемой точности необходимо гарантировать неотрицательность получаемых приближённых решений. Именно подбор параметров  $\rho$  и  $h_0$  в каждой из задач обеспечит выполнение этого свойства. Ниже мы дадим краткое описание рассматриваемых задач, а соответствующие задачи Коши выписаны в Приложении. Для всех задач использовалась представленная в [18] программная технология формирования правой части и матрицы Якоби системы. В этом разделе и в Приложении используются традиционные обозначения  $k_i$  для скоростей элементарных химических реакций (в отличие от векторов стадий из разд. 1).

**Задача 1.** Первая задача описывает процесс пиролиза этана с образованием этилена в проточном реакторе с нагревом газовых компонент излучением CO<sub>2</sub>-лазера [11]. Речь идёт об эндотермической реакции разложения C<sub>2</sub>H<sub>6</sub> при атмосферном давлении и температуре 1000 K с основными продуктами реакции C<sub>2</sub>H<sub>4</sub> (этилен), CH<sub>4</sub> (метан), H<sub>2</sub> и малыми включениями CH<sub>3</sub> (метил), C<sub>2</sub>H<sub>2</sub> (ацетилен), C<sub>3</sub>H<sub>6</sub> (пропилен) — всего 7 реагирующих компонент. В систему уравнений для удобства введена 8-я компонента, которая может быть исключена в силу имеющего место закона сохранения

$$8y_1 + 4y_2 + 7y_3 + 5y_4 + 2y_5 + 3y_6 + 6y_7 + 6y_8 \equiv \text{const}$$

(см. Приложение).

Результаты сравнения времени решения задачи 1 представлены в табл. 10 и на рис. 5. Для данной задачи во всех трёх методах полагалось  $\rho = 10^{-6}$  и  $h_0 = 10^{-6}$ с, т. е. значения параметров, рекомендуемые в [17] для RADAU5.

Рассмотрим вопрос о погрешности решений. В отличие от задачи Робертсона точное решение для вычисления погрешности отсутствует. В связи с этим приведём заданные формулой

Таблица 10

Время решения Задачи 1 (4,2)-, (5,2)-методами и RADAU5

$\varepsilon$	$10^{-7}$	$10^{-6}$	$10^{-5}$	$10^{-4}$	$10^{-3}$	$10^{-2}$
(4,2)-метод	$2.97 \cdot 10^{-3}$	$1.66 \cdot 10^{-3}$	$1.10 \cdot 10^{-3}$	$8.99 \cdot 10^{-4}$	$8.58 \cdot 10^{-4}$	$8.57 \cdot 10^{-4}$
(5,2)-метод	$1.06 \cdot 10^{-3}$	$9.33 \cdot 10^{-4}$	$9.31 \cdot 10^{-4}$	$9.30 \cdot 10^{-4}$	$9.27 \cdot 10^{-4}$	$9.68 \cdot 10^{-4}$
RADAU5	$1.94 \cdot 10^{-3}$	$1.48 \cdot 10^{-3}$	$1.22 \cdot 10^{-3}$	$1.15 \cdot 10^{-3}$	$1.22 \cdot 10^{-3}$	$2.40 \cdot 10^{-3}$

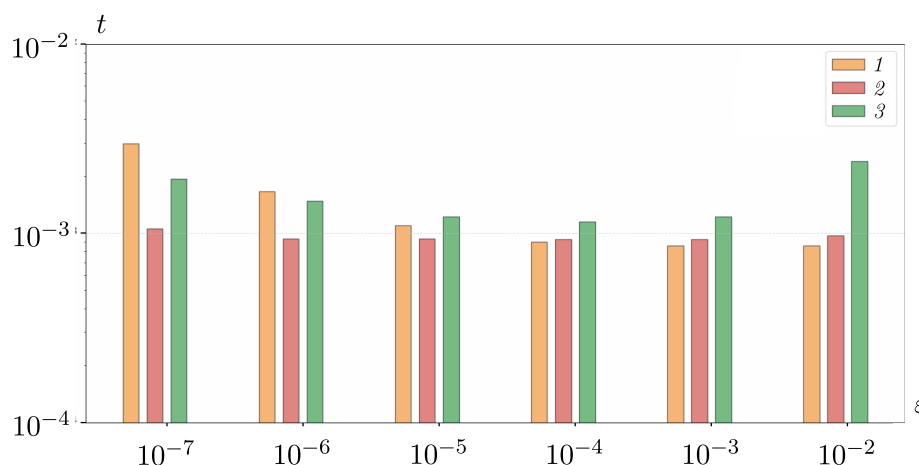


Рис. 5. Время решения Задачи 1 (4,2)-, (5,2)-методами и RADAU5;  
1 — набор 1; 2 — набор 2; 3 — набор 3

(6) значения относительной погрешности  $\varepsilon_{\text{end}} = \varepsilon_{n_{\text{end}}}(j_{n_{\text{end}}})$ , где  $n_{\text{end}}$  — номер слоя, соответствующий конечному моменту времени.

Согласно табл. 11, методы «работают» в заданном диапазоне точности. При этом табл. 10 и рис. 5 иллюстрируют ожидаемую производительность: на низкой точности при  $\varepsilon \geq 10^{-4}$  наиболее экономичным является (4,2)-метод, но проигрывает двум другим методам при высокой точности  $\varepsilon \leq 10^{-5}$ . При  $\varepsilon \leq 10^{-5}$  лучшую производительность показывает (5,2)-метод. При этом для любой задаваемой относительной ошибки (5,2)-метод работает быстрее RADAU5.

**Задача 2.** Рассматривается модель процесса горения водорода  $\text{H}_2/\text{O}_2$ , описанного в работе [12]. В реакциях данной модели участвует 11 реагентов:  $\text{H}$ ,  $\text{O}$ ,  $\text{OH}$ ,  $\text{H}_2$ ,  $\text{O}_2$ ,  $\text{H}_2\text{O}$ ,  $\text{HO}_2$ ,  $\text{H}_2\text{O}_2$ ,  $\text{N}_2$ ,  $\text{Ar}$ ,  $\text{He}$ . Полученная с использованием [18] система уравнений, соответствующая кинетической схеме из статьи [12], приведена в Приложении. Там же указаны константы скоростей элементарных реакций и начальные данные.

Результаты сравнения времени решения задачи 2 представлены в табл. 12 и на рис. 6. Как и для задачи 1, здесь полагалось  $\rho = 10^{-6}$ ,  $h_0 = 10^{-6}$ .

Аналогично табл. 11, в табл. 13 приведены значения вычисляемой ошибки  $\varepsilon_{\text{end}}$  в конечный момент времени. При этом использовались указанные выше значения параметров  $\rho$  и  $h_0$ . Из табл. 13 видно, что вычисленная относительная ошибка не превосходит заданной. Сравнение времени решения показывает преимущество (4,2)-метода при точности, соответствующей значению  $\varepsilon = 10^{-2}$ . При  $\varepsilon \leq 10^{-3}$  наиболее экономичным является (5,2)-метод. Отметим, что являясь методом 5-го порядка точности RADAU5 показывает лучшую по сравнению с (4,2)-методом производительность только при  $\varepsilon \leq 10^{-5}$ .

**Задача 3.** Рассматривается процесс горения малых углеводородов в соответствии с подробным механизмом реакции  $\text{C}/\text{H}/\text{N}/\text{O}$ , предложенным А.А. Конновым и размещённым в Интернете в 2000 году [13]. С тех пор этот механизм использовался во многих исследованиях

Таблица 11

Погрешность  $\varepsilon_{end}$  решения Задачи 1 (4,2)-, (5,2)-методами

$\varepsilon$	$10^{-7}$	$10^{-6}$	$10^{-5}$	$10^{-4}$	$10^{-3}$	$10^{-2}$
(4,2)-метод	$1.09 \cdot 10^{-8}$	$4.33 \cdot 10^{-8}$	$4.11 \cdot 10^{-7}$	$1.04 \cdot 10^{-6}$	$3.09 \cdot 10^{-6}$	$3.09 \cdot 10^{-6}$
(5,2)-метод	$8.94 \cdot 10^{-8}$	$7.86 \cdot 10^{-7}$	$9.94 \cdot 10^{-7}$	$9.94 \cdot 10^{-7}$	$9.94 \cdot 10^{-7}$	$9.94 \cdot 10^{-7}$

Таблица 12

Время решения Задачи 2 (4,2)-, (5,2)-методами и RADAU5

$\varepsilon$	$10^{-7}$	$10^{-6}$	$10^{-5}$	$10^{-4}$	$10^{-3}$	$10^{-2}$
(4,2)-метод	$1.68 \cdot 10^{-2}$	$8.42 \cdot 10^{-3}$	$4.60 \cdot 10^{-3}$	$2.85 \cdot 10^{-3}$	$1.61 \cdot 10^{-3}$	$1.38 \cdot 10^{-3}$
(5,2)-метод	$3.79 \cdot 10^{-3}$	$2.41 \cdot 10^{-3}$	$1.62 \cdot 10^{-3}$	$1.52 \cdot 10^{-3}$	$1.50 \cdot 10^{-3}$	$1.59 \cdot 10^{-3}$
RADAU5	$5.11 \cdot 10^{-3}$	$5.20 \cdot 10^{-3}$	$4.06 \cdot 10^{-3}$	$3.97 \cdot 10^{-3}$	$4.72 \cdot 10^{-3}$	$4.58 \cdot 10^{-3}$

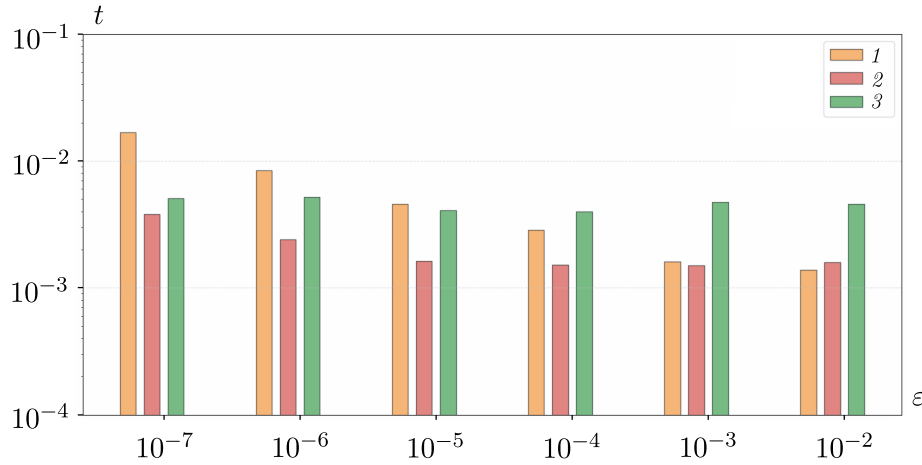


Рис. 6. Время решения Задачи 2 (4,2)-, (5,2)-методами и RADAU5;  
1 — набор 1; 2 — набор 2; 3 — набор 3

— например, при исследовании образования оксида азота NO при горении метано-воздушной смеси  $\text{CH}_4 + \text{H}_2 + \text{O}_2 + \text{N}_2$  [19]. В Приложении приведена задача Коши, соответствующая базовому механизму Коннова [13].

Данная задача оказалась достаточно трудной в смысле подбора приемлемых параметров  $\rho$  и  $h_0$  для демонстрации конкурентоспособности процедур библиотеки CCODE. В приводимых ниже результатах полагалось  $\rho = \rho(\varepsilon)$  и  $h_0 = h_0(\varepsilon)$ . При этом для  $\varepsilon = 10^{-7}, \dots, \varepsilon = 10^{-3}$  имеем  $h_0(\varepsilon) = 10^{-6}$  и  $h_0(10^{-2}) = 10^{-5}$ . Значения  $\rho(\varepsilon)$  указаны в табл. 14. Отметим, что для этих параметров при  $\varepsilon = 10^{-3}$  и  $\varepsilon = 10^{-2}$  не работает (5,2)-метод. Параметры, при которых (5,2)-метод успешно решает задачу 3 для  $\varepsilon = 10^{-3}$  будут приведены отдельно.

На рис. 7 приведены диаграммы, иллюстрирующие табл. 14 с переменным параметром  $\rho$ .

Для использованных выше параметров в табл. 15 приведены значения величины  $\varepsilon_{end}$  в конечный момент времени.

Табл. 15 показывает большую избыточную точность, т. е. решения основными и вложенными методами практически совпадают. При этом наилучшее время демонстрирует (5,2)-метод при заданной относительной ошибке  $\varepsilon = 10^{-4}$  и значениях параметров  $\rho = 10^{-7}$  и  $h_0 = 10^{-6}$ .

И, наконец, приведём значения параметров, при которых (5,2)-метод решает задачу 3 для погрешности  $\varepsilon = 10^{-3}$ . Это  $\rho = 10^{-3}$  и  $h_0 = 10^{-6}$ . Для этих параметров время решения задачи

Таблица 13

Погрешность  $\varepsilon_{\text{end}}$  решения Задачи 2 (4,2)-, (5,2)-методами

$\varepsilon$	$10^{-7}$	$10^{-6}$	$10^{-5}$	$10^{-4}$	$10^{-3}$	$10^{-2}$
(4,2)-метод	$2.58 \cdot 10^{-9}$	$2.22 \cdot 10^{-7}$	$6.67 \cdot 10^{-6}$	$6.57 \cdot 10^{-7}$	$7.05 \cdot 10^{-4}$	$6.50 \cdot 10^{-6}$
(5,2)-метод	$8.54 \cdot 10^{-8}$	$2.20 \cdot 10^{-7}$	$1.16 \cdot 10^{-6}$	$2.12 \cdot 10^{-6}$	$2.41 \cdot 10^{-6}$	$2.41 \cdot 10^{-6}$

Таблица 14

Время решения Задачи 3 (4,2)-, (5,2)-методами и RADAU5.

$\varepsilon$	$10^{-7}$	$10^{-6}$	$10^{-5}$	$10^{-4}$	$10^{-3}$	$10^{-2}$
$\rho$	$10^{-3}$	$10^{-4}$	$10^{-5}$	$10^{-7}$	$10^{-5}$	$10^{-6}$
(4,2)-метод	$2.06 \cdot 10^{-2}$	$2.20 \cdot 10^{-2}$	$2.27 \cdot 10^{-2}$	$3.41 \cdot 10^{-2}$	$5.05 \cdot 10^{-3}$	$6.31 \cdot 10^{-3}$
(5,2)-метод	$1.21 \cdot 10^{-2}$	$1.79 \cdot 10^{-2}$	$1.14 \cdot 10^{-2}$	$5.63 \cdot 10^{-3}$	—	—
RADAU5	$9.29 \cdot 10^{-3}$	$9.07 \cdot 10^{-3}$	$8.65 \cdot 10^{-3}$	$8.73 \cdot 10^{-3}$	$7.26 \cdot 10^{-3}$	$8.11 \cdot 10^{-3}$

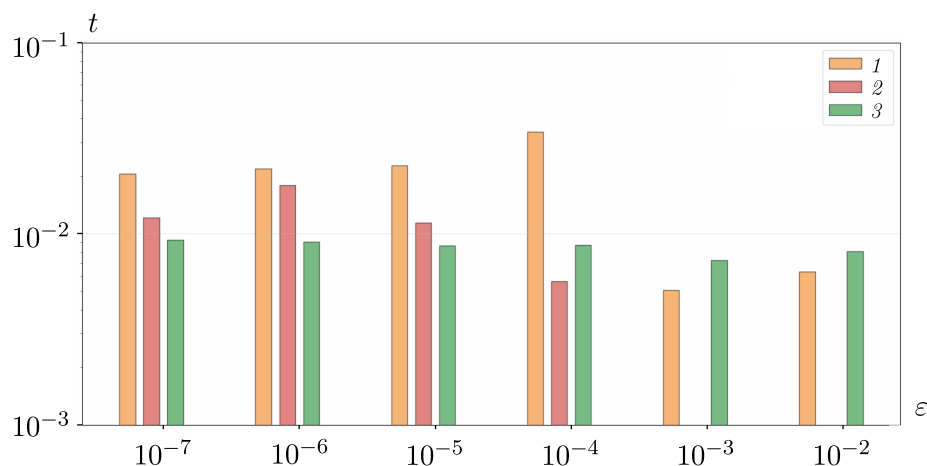


Рис. 7. Время решения Задачи 3 (4,2)-, (5,2)-методами и RADAU5;  
 1 — набор 1; 2 — набор 2; 3 — набор 3

составляет  $2.32 \cdot 10^{-3}$ с, что существенно меньше, чем для (4,2)-метода и RADAU5 при  $\rho = 10^{-5}$  и  $h_0 = 10^{-6}$ . Погрешность при тех же значениях параметров составляет  $\varepsilon_{\text{end}} = 1.4 \cdot 10^{-16}$ . Отметим, что для относительной ошибки  $\varepsilon = 10^{-2}$  найти параметры, при которых (5,2)-метод даёт неотрицательное решение, не удалось.

## 5. Заключительные замечания

Основным результатом данной работы явилось создание библиотеки программ CCODE и демонстрация её функционирования на некоторых задачах химической кинетики. При этом на рассмотренных задачах производительность отдельных функций библиотеки на задаваемой относительной точности порядка  $10^{-4}$  превосходит производительность программы RADAU5. Методической основой библиотеки явились  $L$ -устойчивые с внутренними  $L$ -устойчивыми процедурами (4,2)- и (5,2)-методы 3-го и 4-го порядка точности соответственно. Было проведено исследование по выбору наилучшего набора коэффициентов указанных методов на чрезвычайно жёсткой системе Робертсона, являющейся общепризнанным тестом для методов решения жёстких систем ОДУ. При этом все наборы коэффициентов, обеспечивающие указанные выше порядки точности, включены в библиотеку, и пользователю предоставлена возможность

Таблица 15

Погрешность  $\varepsilon_{\text{end}}$  решения Задачи 3 (4,2)-, (5,2)-методами

$\varepsilon$	$10^{-7}$	$10^{-6}$	$10^{-5}$	$10^{-4}$	$10^{-3}$	$10^{-2}$
$\rho$	$10^{-3}$	$10^{-4}$	$10^{-5}$	$10^{-7}$	$10^{-5}$	$10^{-6}$
(4,2)-метод	$3.7 \cdot 10^{-19}$	$2.5 \cdot 10^{-17}$	$1.1 \cdot 10^{-16}$	$2.1 \cdot 10^{-15}$	$1.2 \cdot 10^{-15}$	$1.9 \cdot 10^{-15}$
(5,2)-метод	$6.6 \cdot 10^{-21}$	$1.9 \cdot 10^{-19}$	$7.3 \cdot 10^{-19}$	$3.4 \cdot 10^{-16}$	—	—

самому выбрать наиболее подходящий набор для конкретной решаемой задачи.

Разработанная библиотека включает ряд возможностей, вскользь упомянутых и даже вообще не описанных в данной статье. В частности, наряду с заданием матрицы Якоби возможно использование процедуры *scoden*, в которой матрица Якоби вычисляется по известной правой части системы. В этом случае пользователь избавлен от довольно трудоёмкой процедуры задания матрицы, но, как указано в [20], теоретический порядок  $(m, 2)$ -методов не превышает 3. Далее, в библиотеке предусмотрен режим “замораживания” матрицы Якоби, когда вычисления на нескольких шагах проводятся без перевычисления матрицы Якоби. Как и при аппроксимации матрицы Якоби, максимальный порядок точности  $(m, 2)$ -методов с замораживанием не превышает 3 (см. [9]).

Развитие библиотеки CCODE связано, главным образом, с её расширением за счёт включения новых  $(m, k)$ -методов. В частности, речь идёт о  $(6, 3)$ -методе 5-го порядка точности, а также  $(3, 2)$ -методах 2-го порядка точности для решения задач с низкой требуемой точностью. Кроме того, возможно расширение библиотеки включением в неё алгоритмов переменного порядка точности без перевычисления матрицы Якоби. Отметим, что такого сорта алгоритм с переходом с  $(4, 2)$ -метода на  $(5, 2)$ -метод и обратно в настоящее время ещё не построен.

## Приложение

### Задача 1 (лазерный пиролиз этана [11]).

$$\begin{aligned}
 \frac{dy_1}{dt} &= -k_0 y_1 - k_1 y_1 y_2 - k_3 y_1 y_6 + k_5 y_4^2 \\
 \frac{dy_2}{dt} &= 2k_0 y_1 - k_1 y_1 y_2 - k_4 y_2 y_5 + k_0 y_8 \\
 \frac{dy_3}{dt} &= k_1 y_1 y_2 \\
 \frac{dy_4}{dt} &= k_1 y_1 y_2 - k_2 y_4 + k_0 y_5 y_6 + k_3 y_1 y_6 - 2k_5 y_4^2 \\
 \frac{dy_5}{dt} &= k_2 y_4 - k_0 y_5 y_6 - k_4 y_2 y_5 + k_0 y_8 + k_5 y_4^2 \\
 \frac{dy_6}{dt} &= k_2 y_4 - k_0 y_5 y_6 - k_3 y_1 y_6 \\
 \frac{dy_7}{dt} &= k_3 y_1 y_6 \\
 \frac{dy_8}{dt} &= k_4 y_2 y_5 - k_0 y_8,
 \end{aligned}$$

где  $0 < t \leq 100$ с, скорости реакций задаются в соответствии с законом Аррениуса

$$k_\alpha = A_\alpha \exp(-E_\alpha/RT),$$

где  $T$  – температура,  $A_\alpha$  – предэкспоненциальный множитель,  $E_\alpha$  – энергия активации,  $R$  – универсальная газовая постоянная. Значения параметров скоростей реакций:

$$\begin{array}{ll}
 A_0 = 10^{16} \text{ с}^{-1}, & E_0/R = 4.27283832 \cdot 10^4 \text{ K} \\
 A_1 = 10^{10} \text{ с}^{-1}, & E_1/R = 5.93449766 \cdot 10^3 \text{ K} \\
 A_2 = 10^{13.5} \text{ с}^{-1}, & E_2/R = 2.01772921 \cdot 10^4 \text{ K} \\
 A_{02} = 10^{10.4} \text{ с}^{-1}, & E_{02}/R = 9.96995607 \cdot 10^2 \text{ K} \\
 A_3 = 10^{11} \text{ с}^{-1}, & E_3/R = 4.74759813 \cdot 10^3 \text{ K} \\
 A_4 = 10^{8.5} \text{ с}^{-1}, & E_4/R = 3.91676846 \cdot 10^3 \text{ K} \\
 A_{04} = 10^{13.9} \text{ с}^{-1}, & E_{04}/R = 1.62605236 \cdot 10^4 \text{ K} \\
 A_5 = 10^{11.5} \text{ с}^{-1}, & E_5/R = 9.96995607 \cdot 10^2 \text{ K}
 \end{array}$$

Начальные условия задаются равенствами:

$$y_1(0) = 12.1875/T, \quad y_2(0) = \dots = y_8(0) = 0.$$

## Задача 2 (реакция $\text{H}_2/\text{O}_2$ [12]).

$$\begin{aligned}
 \frac{dy_1}{dt} = & -k_1 y_1 y_2 - k_2 y_1 y_4 - k_3 y_1 y_5 - k_{024} y_1 y_7 - k_{019} y_1 y_{11} \\
 & + k_{02} y_3 y_5 + k_{03} y_3 y_6 + k_{19} y_3 y_{10} + k_{24} y_3 y_{11} + k_{01} y_2 y_3^2,
 \end{aligned}$$

$$\frac{dy_2}{dt} = 0,$$

$$\begin{aligned}
 \frac{dy_3}{dt} = & 2k_1 y_1 y_2 + k_2 y_1 y_4 + k_3 y_1 y_5 + k_{024} y_1 y_7 + k_{019} y_1 y_{11} + k_{06} y_2 y_6 + k_{08} y_2 y_5 + k_{09} y_2 y_8 \\
 & - k_{02} y_3 y_5 - k_{03} y_3 y_6 - (k_5 + k_{30}) y_3 y_7 - (k_{18} + k_{19}) y_3 y_{10} - (k_{23} + k_{24}) y_3 y_{11} \\
 & + k_{05} y_4 y_5 + k_{018} y_5 y_6 + k_{023} y_5^2 - (k_8 + k_9) y_2 y_3 y_4 - k_6 y_2 y_3 y_5 - 2k_{01} y_2 y_3^2 + k_{030} y_{11},
 \end{aligned}$$

$$\begin{aligned}
 \frac{dy_4}{dt} = & -k_2 y_1 y_4 + 2k_{04} y_2 y_7 + k_{08} y_2 y_5 + k_{09} y_2 y_8 + k_{02} y_3 y_5 + k_5 y_3 y_7 - k_{05} y_4 y_5 - k_7 y_4 y_6 \\
 & - k_{20} y_4 y_{10} - k_{25} y_4 y_{11} + k_{025} y_5 y_7 + k_{020} y_5 y_{11} + k_{07} y_5^2 - (k_8 + k_9) y_2 y_3 y_4 - 2k_4 y_2 y_4^2,
 \end{aligned}$$

$$\begin{aligned}
 \frac{dy_5}{dt} = & k_2 y_1 y_4 - (k_3 + k_{011}) y_1 y_5 + k_{11} y_1 y_8 - k_{08} y_2 y_5 + k_{06} y_2 y_6 + k_{03} y_3 y_6 - (k_{02} + k_{013}) y_3 y_5 \\
 & + k_5 y_3 y_7 + k_{13} y_3 y_8 + k_{18} y_3 y_{10} + 2k_{23} y_3 y_{11} - k_{05} y_4 y_5 + 2k_7 y_4 y_6 + k_{20} y_4 y_{10} + k_{25} y_4 y_{11} \\
 & - (k_{010} + k_{018}) y_5 y_6 - (k_{016} + k_{025}) y_5 y_7 + k_{12} y_5 y_8 - k_{014} y_5 y_9 - (k_{020} + k_{26} + k_{27}) y_5 y_{11} \\
 & + (k_{026} + k_{027}) y_6 y_7 + k_{10} y_6 y_8 + (k_{021} + k_{022}) y_6 y_{11} + k_{16} y_7 y_8 + k_{14} y_8 y_9 + k_8 y_2 y_3 y_4 \\
 & - k_6 y_2 y_3 y_5 - (2k_{07} + k_{012} + 2k_{017} + 2k_{023}) y_5^2 - k_{015} y_5 + k_{15} y_8 + 2k_{17} y_{10},
 \end{aligned}$$

$$\begin{aligned}
 \frac{dy_6}{dt} = & k_3 y_1 y_5 - k_{06} y_2 y_6 - k_{03} y_3 y_6 + k_{18} y_3 y_{10} - k_7 y_4 y_6 - k_{018} y_5 y_6 + (k_{21} + k_{22}) y_5 y_{10} \\
 & + (k_{26} + k_{27}) y_5 y_{11} - (k_{026} + k_{027}) y_6 y_7 - (k_{021} + k_{022}) y_6 y_{11} + k_6 y_2 y_3 y_5,
 \end{aligned}$$



$$\begin{aligned}\frac{dy_7}{dt} = & -k_{024}y_1y_7 - k_{04}y_2y_7 - (k_5 + k_{30})y_3y_7 + k_{24}y_3y_{11} + k_{05}y_4y_5 + k_{25}y_4y_{11} - k_{025}y_5y_7 \\ & + (k_{26} + k_{27})y_5y_{11} - k_{027}y_6y_7 - (k_{028} + k_{029})y_7y_{10} + k_{28}y_{10}^2 + k_{29}y_{11}^2 + k_4y_1y_4^2 + k_{030}y_{11},\end{aligned}$$

$$\begin{aligned}\frac{dy_8}{dt} = & -k_{11}y_1y_8 + k_{011}y_1y_5 - k_{09}y_2y_8 - k_{13}y_3y_8 + k_{013}y_3y_5 \\ & + k_{010}y_5y_6 + k_{016}y_5y_7 - k_{12}y_5y_8 + k_{014}y_5y_9 - k_{10}y_6y_8,\end{aligned}$$

$$\frac{dy_9}{dt} = 0,$$

$$\begin{aligned}\frac{dy_{10}}{dt} = & k_{019}y_1y_{11} - (k_{18} + k_{19})y_3y_{10} - k_{20}y_4y_{10} + k_{018}y_5y_6 - (k_{21} + k_{22})y_5y_{10} + k_{021}y_6y_{11} \\ & + (k_{020} + k_{022})y_5y_{11} - (k_{028} + k_{029})y_7y_{10} + (k_{28} + k_{29})y_{11}^2 + k_{017}y_5^2 - k_{17}y_{10},\end{aligned}$$

$$\begin{aligned}\frac{dy_{11}}{dt} = & k_{024}y_1y_7 - k_{019}y_1y_{11} + k_{30}y_3y_7 + k_{19}y_3y_{10} - (k_{23} + k_{24})y_3y_{11} + k_{20}y_4y_{10} - k_{25}y_4y_{11} \\ & + k_{025}y_5y_7 + (k_{21} + k_{22})y_5y_{10} - y_5y_{11} - (k_{020} + k_{26} + k_{27})y_5y_{11} + (k_{026} + k_{027})y_6y_7 \\ & - (k_{021} + k_{022})y_6y_{11} + 2(k_{028} + k_{029})y_7y_{10} + k_{023}y_5^2 - 2(k_{28} + k_{29})y_{10}^2 - k_{030}y_{11},\end{aligned}$$

где  $0 < t \leq 200$ с. Значения скоростей реакций (1/с):

$k_1 = 1.21 \cdot 10^{-17},$	$k_{01} = 1.09 \cdot 10^{15},$	$k_{16} = 7.83 \cdot 10^{13},$	$k_{016} = 9.25 \cdot 10^{-16},$
$k_2 = 2.18 \cdot 10^{10},$	$k_{02} = 2.81 \cdot 10^{10},$	$k_{17} = 4.38 \cdot 10^{-1},$	$k_{017} = 9.59 \cdot 10^{12},$
$k_3 = 2.88 \cdot 10^{11},$	$k_{03} = 3.29 \cdot 10^7,$	$k_{18} = 1.39 \cdot 10^{12},$	$k_{018} = 3.86 \cdot 10^{-11},$
$k_4 = 2.33 \cdot 10^{14},$	$k_{04} = 5.21 \cdot 10^{-22},$	$k_{19} = 2.02 \cdot 10^{11},$	$k_{019} = 3.61 \cdot 10^5,$
$k_5 = 1.76 \cdot 10^9,$	$k_{05} = 1.13 \cdot 10^{13},$	$k_{20} = 2.70 \cdot 10^{11},$	$k_{020} = 6.22 \cdot 10^5,$
$k_6 = 7.14 \cdot 10^{16},$	$k_{06} = 9.09 \cdot 10^{-20},$	$k_{21} = 1.38 \cdot 10^{12},$	$k_{021} = 2.84 \cdot 10^2,$
$k_7 = 9.89 \cdot 10^7,$	$k_{07} = 1.11 \cdot 10^{12},$	$k_{22} = 4.08 \cdot 10^{11},$	$k_{022} = 8.37 \cdot 10,$
$k_8 = 6.73 \cdot 10^{15},$	$k_{08} = 9.65 \cdot 10^{-17},$	$k_{23} = 5.73 \cdot 10^{13},$	$k_{023} = 7.77,$
$k_9 = 2.05 \cdot 10^{11},$	$k_{09} = 2.48 \cdot 10^8,$	$k_{24} = 9.22 \cdot 10^{12},$	$k_{024} = 1.51 \cdot 10^{-4},$
$k_{10} = 2.91 \cdot 10^{14},$	$k_{010} = 3.44 \cdot 10^{-15},$	$k_{25} = 3.25 \cdot 10^{13},$	$k_{025} = 6.88 \cdot 10^{-4},$
$k_{11} = 1.07 \cdot 10^{14},$	$k_{011} = 1.27 \cdot 10^{-15},$	$k_{26} = 1.54 \cdot 10^{13},$	$k_{026} = 2.89 \cdot 10^{-8},$
$k_{12} = 2.75 \cdot 10^{14},$	$k_{012} = 3.25 \cdot 10^{-15},$	$k_{27} = 1.74 \cdot 10^{11},$	$k_{027} = 3.27 \cdot 10^{-10},$
$k_{13} = 3.91 \cdot 10^{13},$	$k_{013} = 4.62 \cdot 10^{-16},$	$k_{28} = 3.57 \cdot 10^{10},$	$k_{028} = 3.28 \cdot 10^{-1},$
$k_{14} = 8.65 \cdot 10^{10},$	$k_{014} = 1.02 \cdot 10^{-18},$	$k_{29} = 5.23 \cdot 10^{11},$	$k_{029} = 4.80,$
$k_{15} = 1.45 \cdot 10^6,$	$k_{015} = 1.71 \cdot 10^{-23},$	$k_{30} = 8.30 \cdot 10^{13},$	$k_{030} = 5.62 \cdot 10^{-2}.$

Начальные условия:

$$\begin{aligned}y_1(0) &= 2.32 \cdot 10^{-3}, \quad y_7(0) = 1.16 \cdot 10^{-3}, \quad y_9(0) = 1.39 \cdot 10^{-1}, \\ y_2(0) &= \dots = y_6(0) = y_8(0) = y_{10}(0) = y_{11}(0) = 0.\end{aligned}$$

**Задача 3 (горение малых углеводородов [13]).**

$$\begin{aligned}\frac{dy_1}{dt} = & -k_{16}y_1y_6 - (k_{20} + k_{21} + k_{22})y_1y_9 - (k_{29} + k_{30})y_1y_{11} - k_7y_2y_8 + k_{15}3y_3y_5 \\ & + k_{17}3y_3y_7 + k_8y_8^2 - k_{11}y_1y_6^2 - 2k_1y_2y_1^2 - 2k_2y_3y_1^2 - 2k_3y_4y_1^2 - 2k_4y_1^3 - k_6y_1y_2y_5 \\ & - k_9y_1y_2y_6 - k_{12}y_1y_6y_8 - k_{10}y_1y_6y_{10},\end{aligned}$$

$$\frac{dy_2}{dt} = 0,$$

$$\frac{dy_3}{dt} = k_{22}y_1y_9 + k_{29}y_1y_{11} - k_{15}y_3y_5 - k_{23}y_3y_6 - k_{17}y_3y_7 + k_1y_2y_1^2 + k_2y_3y_1^2 + k_3y_4y_1^2 + k_4y_1^3,$$

$$\frac{dy_4}{dt} = 0,$$

$$\frac{dy_5}{dt} = k_{16}y_1y_6 + k_{21}y_1y_9 - k_{15}y_3y_5 - k_{19}y_5y_9 - k_{31}y_5y_{11} + k_{18}y_7^2 - 2k_5y_2y_5^2 - k_6y_1y_2y_5,$$

$$\begin{aligned}\frac{dy_6}{dt} = & -k_{16}y_1y_6 + k_{22}y_1y_9 - k_{23}y_3y_6 + k_{19}y_6y_9 + (k_{24} + k_{25})y_7y_9 + (k_{26} + k_{27})y_9^2 - k_{11}y_1y_6^2 \\ & + k_5y_2y_5^2 + k_{28}y_2y_9^2 - k_9y_1y_2y_6 - k_{12}y_1y_6y_8 - k_{10}y_1y_6y_{10},\end{aligned}$$

$$\begin{aligned}\frac{dy_7}{dt} = & k_{16}y_1y_6 + 2k_{20}y_1y_9 + k_{30}y_1y_{11} + k_{15}y_2y_5 + k_7y_2y_8 + 2k_{23}y_3y_6 - k_{17}y_3y_7 \\ & + k_{19}y_5y_9 + k_{31}y_5y_{11} - (k_{24} + k_{25})y_7y_9 - (k_{32} + k_{33})y_7y_{11} - 2k_{18}y_7^2 \\ & + k_8y_8^2 - 2k_{13}y_2y_7^2 - 2k_{14}y_8y_7^2 + k_6y_1y_2y_5,\end{aligned}$$

$$\begin{aligned}\frac{dy_8}{dt} = & k_{21}y_1y_9 + k_{30}y_1y_{11} - k_7y_2y_8 + k_{17}y_3y_7 + (k_{24} + k_{25})y_7y_9 \\ & + (k_{32} + k_{33})y_7y_{11} + k_{18}y_7^2 - k_8y_8^2,\end{aligned}$$

$$\begin{aligned}\frac{dy_9}{dt} = & -(k_{20} + k_{21} + k_{22})y_1y_9 + k_{29}y_1y_{11} - k_{19}y_5y_9 + k_{31}y_5y_{11} - (k_{24} + k_{25})y_7y_9 + k_{11}y_1y_6^2 \\ & + (k_{32} + k_{33})y_7y_{11} - 2(k_{26} + k_{27} + k_{28})y_9^2 + k_9y_1y_2y_6 + k_{12}y_1y_6y_8 + k_{10}y_1y_6y_{10},\end{aligned}$$

$$\frac{dy_{10}}{dt} = 0,$$

$$\begin{aligned}\frac{dy_{11}}{dt} = & -(k_{29} + k_{30})y_1y_{10} - k_{31}y_4y_{10} - (k_{32} + k_{33})y_6y_{10} \\ & + (k_{26} + k_{27})y_8^2 + k_{13}y_1y_6^2 + k_{28}y_1y_8^2 + k_{14}y_7y_6^2,\end{aligned}$$

где  $0 < t \leq 10$ с. Значения скоростей реакций (1/с):

$$\begin{array}{lll}k_1 = 2.10 \cdot 10^{15}, & k_{12} = 2.89 \cdot 10^{13}, & k_{23} = 2.95 \cdot 10^2, \\ k_2 = 3.06 \cdot 10^{15}, & k_{13} = 9.79 \cdot 10^{12}, & k_{24} = 3.46 \cdot 10^{13}, \\ k_3 = 2.84 \cdot 10^{15}, & k_{14} = 9.79 \cdot 10^{12}, & k_{25} = 1.67 \cdot 10^{13},\end{array}$$

$$\begin{array}{lll}
k_4 = 3.20 \cdot 10^{15}, & k_{15} = 3.37 \cdot 10^{10}, & k_{26} = 1.91 \cdot 10^{12}, \\
k_5 = 3.00 \cdot 10^{14}, & k_{16} = 5.07 \cdot 10^{11}, & k_{27} = 3.22 \cdot 10^{11}, \\
k_6 = 2.03 \cdot 10^{16}, & k_{17} = 3.74 \cdot 10^{11}, & k_{28} = 1.38 \cdot 10^{15}, \\
k_7 = 3.26, & k_{18} = 7.59 \cdot 10^{10}, & k_{29} = 4.38 \cdot 10^{11}, \\
k_8 = 1.25 \cdot 10, & k_{19} = 1.91 \cdot 10^{13}, & k_{30} = 2.75 \cdot 10^{12}, \\
k_9 = 4.76 \cdot 10^{13}, & k_{20} = 1.38 \cdot 10^{14}, & k_{31} = 2.52 \cdot 10^{11}, \\
k_{10} = 4.76 \cdot 10^{13}, & k_{21} = 1.45 \cdot 10^{12}, & k_{32} = 1.71 \cdot 10^{12}, \\
k_{11} = 4.76 \cdot 10^{13}, & k_{22} = 5.01 \cdot 10^{13}, & k_{33} = 4.15 \cdot 10^{13},
\end{array}$$

Начальные условия:

$$\begin{aligned}
y_3(0) &= 4.88 \cdot 10^{-3}, & y_6(0) &= 2.44 \cdot 10^{-3}, & y_{10}(0) &= 2,93 \cdot 10^{-2}, \\
y_1(0) &= y_2(0) = y_4(0) = y_5(0) = y_7(0) = y_8(0) = y_9(0) = y_{11}(0) = 0.
\end{aligned}$$

### ФИНАНСИРОВАНИЕ РАБОТЫ

Работа выполнена в рамках научной программы Национального центра физики и математики, направление №2 «Математическое моделирование на супер-ЭВМ экса- и зеттапроизводительности. Этап 2023–2025». Других источников финансирования проведения или руководства данным конкретным исследованием не было.

### КОНФЛИКТ ИНТЕРЕСОВ

Авторы данной работы заявляют, что у них нет конфликта интересов.

### ЛИТЕРАТУРА

1. Хайпер Э., Ваннер Г. Решение обыкновенных дифференциальных уравнений. Жёсткие и дифференциально-алгебраические задачи. М.: Мир, 1999. Solving Ordinary Differential Equations II. Stiff and Differential-Algebraic Problems. Berlin – Heidelberg: Springer-Verlag, 1991. /
2. Деулин А.А., Глазунов В.А., и др. Подходы к моделированию кинетики химических реакций в пакете программ «ЛОГОС» // Сб. тезисов XXV Международной конференции по химическим реакторам (ХИМПРЕАКТОР-25). Новосибирск, 2023. С. 55.
3. Новиков Е.А., Шитов Ю.А., Шокин Ю.И. Одношаговые безытерационные методы решения жёстких систем // Докл. АН СССР. 1988. Т. 301, № 6. С. 1310–1314.
4. Новиков Е.А., Шитов Ю.А., Шокин Ю.И. О классе  $(m, k)$ -методов решения жёстких систем // Ж. вычисл. матем. и матем. физ. 1989. Т. 29, № 2. С. 194–201.
5. Новиков Е.А. Исследование  $(m, 2)$ -методов решения жёстких систем // Вычислительные технологии. 2007. Т. 12, № 5. С. 103–115.
6. Новиков Е.А., Двинский А.Л.  $(4, 2)$ -метод третьего порядка для решения жёстких систем // Вычислительные технологии. 2001. Т. 6. Ч. 2. Спец. выпуск. С. 470–474.
7. Новиков Е.А., Двинский А.Л.  $(5, 2)$ -метод с внутренней  $L$ -устойчивостью для жёстких систем // Вестник КрасГАУ. Ресурсосберегающие технологии. 2005. № 3. С. 95–100.
8. Новиков Е.А. Оценка глобальной ошибки  $A$ -устойчивых методов решения жёстких систем // Докл. РАН. 1995. Т. 343, № 4. С. 452–455.
9. Двинский А.Л. Исследование  $(m, k)$ -методов с  $L$ -устойчивыми промежуточными схемами для решения жёстких систем // Дис... канд. физ.-мат. наук. Красноярск, 2004.
10. Robertson H.H. The solution of a set of reaction rate equations // Numerical Analysis, An Introduction, J. Walsh, ed. New York: Academic Press, 1966. P. 178–182.

11. *Snytnikov V.N, Mischenkoa T.I., Snytnikov V.I.N., Chernykh I.G.* A reactor for the study of homogeneous processes using laser radiation energy // *Chemical Engrg. J.* 2009. V. 150. P. 231–236.
12. *Burke M.P., Chaos M., Ju Y., Dryer F.L., Klippenstein S.J.* Comprehensive H<sub>2</sub>/O<sub>2</sub> kinetic model for high-pressure combustion // *Int. J. of Chemical Kinetics*. 2011. V. 44, Iss. 7. P. 444–474.
13. *Konnov A.A.* Detailed reaction mechanism for small hydrocarbons combustion. Release 0.5 // <http://homepages.vub.ac.be/~akonnov/>, 2000.
14. *Radhakrishnan K., Hindmarsh A.C.* Description and Use of LSODE, the Livermore Solver for Ordinary Differential Equations // NASA Reference Publication 1327, Lawrence Livermore National Laboratory Report UCRL-ID-113855, 1993.
15. *Brown P.N., Byrne G.D., Hindmarsh A.C.* VODE, A Variable- Coefficient ODE Solver // *SIAM J. Sci. Stat. Comput.* 1989. V. 10. P. 1038–1051.
16. *Gear C.W.* The automatic integration of ordinary differential equations // *Communications of the ACM*. 1971. V. 14, N 3. P. 176–179.
17. *Hairer E.* Testset for Stiff ODEs // <https://www.unige.ch/~hairer/testset/testset.html>.
18. *Черных И.Г.* Алгоритмический и программный инструментарий для численного решения прямых задач химической кинетики с использованием супер-ЭВМ // Дис. ... канд. физ.-мат. наук. Новосибирск, 2006.
19. *Coppens F.H.V. , De Ruyck J., Konnov A.A.* The effects of composition on burning velocity and nitric oxide formation in laminar premixed flames of CH<sub>4</sub> + H<sub>2</sub> + O<sub>2</sub> + N<sub>2</sub> // *Combustion and Flame*. 2007. V. 149. P. 409–417.
20. *Новиков Е.А.* Аппроксимация матрицы Якоби в  $(m, 2)$ -методах решения жёстких задач // *Журн. вычисл. математики и матем. физики*. 2011. Т. 51, № 12. С. 2194–2208.

UDC 519.622.2

## CCODE LIBRARY OF PROGRAMS FOR SOLVING STIFF SYSTEMS OF ORDINARY DIFFERENTIAL EQUATIONS

© 2025 K. P. Saktoeva<sup>a</sup>, Yu. M. Laevsky<sup>b</sup>

*Institute of Computational Mathematics and Mathematical Geophysics SB RAS  
Prosp. Acad. Lavrentieva, 6, Novosibirsk 630090, Russia*

E-mails: <sup>a</sup>saktoeva33@gmail.com, <sup>b</sup>laev@labchem.sccc.ru

Received 24.05.2025, revised 17.06.2025, accepted 17.09.2025

**Abstract.** The paper deals the software library for solving stiff systems of ordinary differential equations, based on variable-step integration (m,k)-methods and tested on several systems from chemical kinetics area. The paper focuses on selecting the most efficient schemes and comparing their performance with the widely used RADAU5 program.

**Keywords:** software library, stiff systems, one-step methods, chemical kinetics, numerical experiment.

DOI: 10.33048/SIBJIM.2025.28.306

### REFERENCES

1. Hairer E., Wanner G. Solving Ordinary Differential Equations II. Stiff and Differential-Algebraic Problems. Berlin – Heidelberg: Springer-Verlag, 1991.
2. Deulin A.A., Glazunov V.A., i dr. Podhody k modelirovaniyu kinetiki himicheskikh reakciy v pakete programm “LOGOS” [Approaches to modeling the kinetics of chemical reactions in the LOGOS software package]. *Sb. tezisov XXV Mejdunarodnoy konferencii po himicheskim reaktoram (HIMREAKTOR-25)* [Abstracts of the XXV International Conference on Chemical Reactors (KHIMREAKTOR-25)]. Novosibirsk, 2023, pp. 55 (in Russian).
3. Novikov E.A., Shitov Yu.A., Shokin Yu.I. Odnoshagovye beziteracionnye metody resheniya zhestkih sistem [Single-step, non-iterative methods for solving stiff systems]. *Dokl. AN USSR*, 1988, Vol. 301, No. 6, pp. 1310–1314 (in Russian).
4. Novikov E.A., Shitov Yu.A., Shokin Yu.I. O klasse (m,k)-metodov resheniya zhestkih sistem [A class of (m, k)-methods for solving stiff systems. *Zhurn. Vychisl. Matemat. i Matemat. Fiziki*. [J. Calculation. Math. Math. Phys.], 1989, Vol. 29, No. 2, pp. 194–201 (in Russian).
5. Novikov E.A. Issledovanie (m,2)-metodov resheniya zhestkih sistem [Investigation of (m,2)-methods for solving stiff systems]. *Vychisl. Tekhnol.* [Comput. Technol.], 2007, Vol. 12, No. 5, pp. 103–115 (in Russian).
6. Novikov E.A., Dvinskiy A.L. (4, 2)-metod tretogo poryadka dlya resheniya zhestkih sistem [The third-order (4,2) method for solving stiff systems]. *Vychisl. Tekhnol.* [Comput. Technol.], 2001, Vol. 6, Part 2, Spec. Vypusk, pp. 470–474 (in Russian).
7. Novikov E.A., Dvinskiy A.L. (5, 2)-metod s vnutrenney L-ustoychivostyu dlya zhestkih sistem [The (5, 2) method with internal L-stability for stiff systems]. *Vestn. KrasGAU. Resursosberegayuschie Tehnologii* [Bull. KrasGAU. Resource-Saving Technol.], 2005, No. 3, pp. 95–100 (in Russian).
8. Novikov E.A. Ocenka globalnoy oshibki A-ustoychivyh metodov resheniya zhestkih sistem [Global error estimation of A-stable methods for solving stiff system]. *Dokl. RAN*, 1995, Vol. 43, No. 4, pp. 452–455 (In Russian).

9. Dvinskiy A.L. Issledovanie  $(m, 2)$ -metodov s L-ustoychivymi promezhutochnymi shemami dlya resheniya zhestkih sistem [A study of  $(m, k)$  methods with L-stable intermediate schemes for solving stiff systems]. Diss. ... kand. fiz.-mat. nauk. Krasnoyarsk, 2004 (in Russian).
10. Robertson H.H. The solution of a set of reaction rate equations. *Numerical Analysis, An Introduction*, J. Walsh, ed. New York: Academic Press, 1966, pp. 178–182.
11. Snytnikov V.N., Mischenkoa T.I., Snytnikov V.I., Chernykh I.G. A reactor for the study of homogeneous processes using laser radiation energy. *Chemical Engrg. J.*, 2009, Vol. 150, pp. 231–236.
12. Burke M.P., Chaos M., Ju Y., Dryer F.L., Klippenstein S.J. Comprehensive H<sub>2</sub>/O<sub>2</sub> kinetic model for high-pressure combustion. *Inter. J. Chemical Kinetics*, 2011, Vol. 44, Iss. 7, pp. 444–474.
13. Konnov A.A. Detailed reaction mechanism for small hydrocarbons combustion. Release 0.5, 2000, <http://homepages.vub.ac.be/~akonnov/>
14. Radhakrishnan K., Hindmarsh A.C. Description and Use of LSODE, the Livermore Solver for Ordinary Differential Equations. NASA Reference Publication 1327, Lawrence Livermore National Laboratory Report UCRL-ID-113855, 1993.
15. Brown P.N., Byrne G.D., Hindmarsh A.C. VODE, A Variable- Coefficient ODE Solver. *SIAM J. Sci. Stat. Comput.*, 1989, Vol. 10, pp. 1038–1051.
16. Gear C.W. The automatic integration of ordinary differential equations. *Communications of the ACM*, 1971, Vol. 14, No. 3, pp. 176–179.
17. Hairer E. Testset for Stiff ODEs. <https://www.unige.ch/~hairer/testset/testset.html>.
18. Chernykh I.G. Algoritmicheskiy i programmnyy instrumentariy dlya chislennogo resheniya pryamykh zadach himicheskoy kinetiki s ispolzovaniem super-komputer [Algorithmic and software tools for the numerical solution of direct problems of chemical kinetics using supercomputers]. Diss. ... kand. fiz.-mat. nauk. Novosibirsk, 2006 (in Russian).
19. Coppens F.H.V. , De Ruyck J., Konnov A.A. The effects of composition on burning velocity and nitric oxide formation in laminar premixed flames of CH<sub>4</sub> + H<sub>2</sub> + O<sub>2</sub> + N<sub>2</sub>. *Combustion and Flame*, 2007, Vol. 149, pp. 409–417.
20. Novikov E.A. Approksimatsiya matricy Jacoby in  $(m, 2)$ -metodah resheniya zhestkih zadach [Approximation of the Jacobian matrix in  $(m, 2)$ -methods for solving stiff problems]. *Zhurn. Vychisl. Matematiki i Mat. Fiziki* [J. Calculation. Math. Math. Phys.], 2011, Vol. 51, No. 12, pp. 2194–2208 (in Russian).