

СОВРЕМЕННЫЕ ЧИСЛЕННЫЕ МЕТОДЫ ИНТЕГРИРОВАНИЯ СВЕРХЖЕСТКИХ ОДУ

Власов Сергей Александрович¹

¹Магистр техники и технологии по направлению «Системный анализ и управление»;

ГОУ ВПО «Международный Университет природы, общества и человека «Дубна»,
Институт системного анализа и управления;

141980, Московская обл., г. Дубна, ул. Университетская, 19;

e-mail: vlasovs2@yandex.ru.

Работа посвящена исследованию современных методов интегрирования задачи Коши для сверхжестких систем обыкновенных дифференциальных уравнений. К современным методам были отнесены: полностью неявные и диагонально неявные методы Рунге-Кутты, методы Розенброка, коллокационные методы. Поиск решения производится на основе вычислений повышенной точности.

Часто требуется, чтобы решение было непрерывным. В отличие от разностного решения непрерывное можно вычислять между узлами основной сетки интегрирования. Для того чтобы получился непрерывный метод можно воспользоваться интерполяцией, или попытаться построить решение на основе стадий того или иного метода.

Было выполнено экспериментальное сравнение десятка различных методов, и полученные результаты представлены в виде графиков зависимостей "настраиваемая точность - точность полученного решения" и "точность полученного решения - трудоемкость"

В работу входит описание методов поиска коэффициентов непрерывных методов Розенброка до точности в 30 и более значащих цифр.

Ключевые слова: сверхжесткие обыкновенные дифференциальные уравнения, непрерывные методы Розенброка, вычисления повышенной точности.

MODERN NUMERICAL METHODS FOR SUPER STIFF ODE

Vlasov Sergey Aleksandrovich¹

¹Master of technology and technologies;

International university of the nature, society and man «Dubna», Institute of system analysis and management;

141980, Dubna, Moscow reg., Universitetskaya str., 19;

e-mail: vlasovs2@yandex.ru.

Work is dedicated to study of the modern methods for super stiff ordinary differential equations. There are FIRK and DIRK methods, Rosenbrock methods, and collocation methods. Searching for the solution is produced on base of high precision computations.

It is often required that solution was continuous. Unlike discrete continuous solution can be calculated between nodes of the main net. To get continuous method we can use interpolation, or to try to build solution on base of stage of that or other method.

Experimental comparison dozen different methods was executed, and got results are presented in the manner of graph of the dependencies "adjusted accuracy - accuracy of the got solution" and "accuracy of the got solution - computations time".

The description of the methods of searching for factor solution multirate Rosenbrock methods enters in work before accuracy in 30 and more the significant digits.

Keywords: super stiff ordinary differential equations, multirate Rosenbrock methods, high precision computations.

Введение

Задача Коши для систем обыкновенных дифференциальных уравнений может быть записана в виде:

$$\begin{cases} \frac{d}{dt} \bar{u}(t) = \bar{F}[t, \bar{u}(t)], & t \in [0, T], \\ \bar{u}(0) = \bar{u}_0 \end{cases}, \quad (1)$$

где $\bar{F} : R \times R^n \rightarrow R^n$, $\bar{u}(t) \in R^n$ — искомая вектор-функция; $\bar{u}_0 \in R^n$ — начальные условия.

Часто требуется, чтобы решение было непрерывным. В отличие от разностного решения непрерывное можно вычислять между узлами основной сетки интегрирования. Для того чтобы получился непрерывный метод можно воспользоваться интерполяцией, или попытаться построить решение на основе стадий того или иного метода. Непрерывное решение необходимо, например, при решении ОДУ с запаздываниями.

Задача Коши для систем дифференциальных уравнений с запаздываниями (DDE — delay differential equations) может быть записана в виде:

$$\begin{cases} \frac{d}{dt} \bar{u}(t) = \bar{F}[t, \bar{u}(\beta_1), \bar{u}(\beta_2), \dots, \bar{u}(\beta_k)], & t \in [0, T] \\ \bar{u}(t) = \bar{\varphi}(t), & t \in [-\theta, 0] \end{cases}, \quad (2)$$

где β_i ($i=1..k$) — параметры запаздывания, θ — максимальное запаздывание; $t - \theta \leq \beta_1, \beta_2, \dots, \beta_k \leq t$;

$\bar{F} : R \times (R^n)^k \rightarrow R^n$, $\bar{u}(t) \in R^n$ — искомая вектор-функция; $\bar{\varphi}(t) \in R^n$ — начальные условия.

При поиске новых методов Розенброка часто можно столкнуться с проблемой, когда коэффициенты приведены с малым числом значащих цифр, рассчитанным на вычисления со стандартной точностью, то есть 15-20 значащих цифр. При этом приходится обращаться к условиям порядка и устойчивости, чтобы уточнить коэффициенты методов. В данной работе будет описан метод уточнения коэффициентов, а так же даны сами уточненные коэффициенты некоторых методов.

Тестирование современных методов было проведено на сверхжестких задачах без запаздываний.

1. Современные вычисления

Многие вычисления удобно делать в какой-нибудь системе компьютерной математики, такой как Maple. Система Maple предназначена для символьных вычислений, обладает развитыми графическими средствами, имеет собственный язык программирования.

Большинство поисков решений систем уравнений было произведено именно в Maple. Кроме того, он позволяет производить вычисления произвольной точности с различным количеством значащих цифр.

Так же активно использовался язык программирования C++ и C++ Builder 6 как программная среда.

1.1. Длинная арифметика

Основная масса библиотек длинной арифметики написана под Unix. К ним относится и библиотеки gmp (GNU Multi-Precision Library) и mpfr (multiple-precision floating-point computations with correct rounding), которые позволяют делать вычисления над числами произвольной точности. Библиотека gmp считается самой быстрой библиотекой чисел произвольной точности. Но нам нужна для тестирования методов решения ОДУ библиотека под Windows.

Под Windows не существует библиотек подобных gmp и mpfr. Но есть техника, позволяющая считать числа точности double double [11](30 значащих цифр) и quad double [12](60 значащих цифр). Числа точности double double работают даже быстрее чем числа mpfr той же точности. В литературе используется в основе 64-х битный тип double, но его можно заменить 80-ю битным типом long double (extended), что даст больше значащих цифр. При этом нужно изменить константу разбиения с 134217729 на 8589934593, а так же заменить некоторые формулы (для экспоненты, синуса и гамма-функции) на более точные.

1.2. Получение условий порядка.

Для получения условий порядка была написана программа позволяющая: раскрывать скобки, приводить подобные слагаемые и группировать слагаемые относительно матричных операторов. Риск ошибиться при этом значительно уменьшается. Так были получены условия порядка для всех методов Розенброка включенных в данную работу.

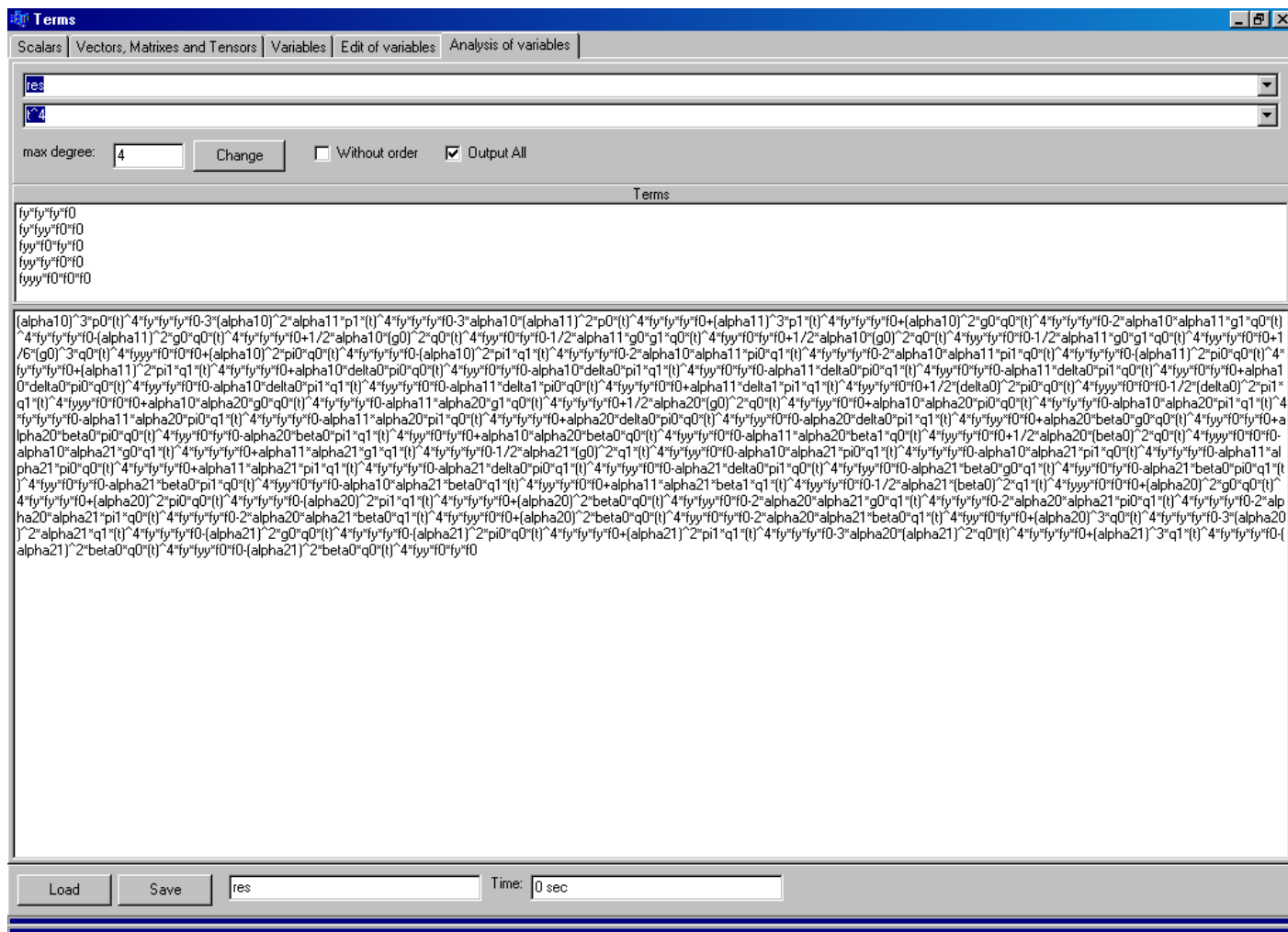


Рисунок 1. Программа получения условий порядка.

1.3. Метод Ньютона для решения некорректных задач.

Имеются в виду задачи нахождения решения системы нелинейных уравнений, в которых количество переменных превышает количество уравнений. Пусть дана система уравнений:

$$\begin{cases} F_1(x_1, x_2, \dots, x_n) = 0 \\ F_2(x_1, x_2, \dots, x_n) = 0 \\ \dots \\ F_k(x_1, x_2, \dots, x_n) = 0 \end{cases}, \quad k < n \quad (3)$$

Эту систему нужно дополнить недостающими уравнениями, например, вида $0=0$. После этого, как обычно в методе Ньютона выбирается начальное приближение и вычисляется матрица Якоби. Матрица Якоби вырождена и имеет определитель равный нулю, поэтому мы не можем перейти к следующей точке по стандартной схеме:

$$\frac{\partial \bar{F}}{\partial \bar{x}}(\bar{x}_i) \cdot \Delta \bar{x}_i = -\bar{F}(\bar{x}_i), \quad (4)$$

$$\bar{x}_{i+1} = \bar{x}_i + \Delta \bar{x}_i$$

где $\frac{\partial \bar{F}}{\partial \bar{x}}(\bar{x}_i)$ — матрица Якоби.

Вместо этого воспользуемся регуляризацией системы (4) по Тихонову. Пусть матрица Якоби равна A , тогда приближенное решение (4) можно найти из следующей системы уравнений:

$$(A^T \cdot A + \lambda \cdot E) \cdot v = A^T \cdot f + \lambda \cdot v^* \quad (5)$$

где λ — параметр регуляризации, E — единичная матрица, $v = \Delta \bar{x}_i$, $f = -\bar{F}(\bar{x}_i)$, v^* — желаемое $\Delta \bar{x}_i$

В нашем случае $v^* = 0$. Параметр регуляризации λ влияет на скорость сходимости получившегося метода Ньютона, чем он больше, тем сходимость медленнее и может вообще остановиться. С другой стороны, если взять очень маленький параметр, то система (5) может дать очень крупные значения и метод Ньютона разойдется. Возможное значение параметра $\lambda = 10^{-12}$.

1.4. Непрерывный метод Розенброка на основе стадий

Чтобы метод был непрерывным, добавляют дополнительные условия порядка относительно параметра θ , зависящего от времени [5]. Решение ищется в виде:

$$y(\theta) = \sum_{i=1}^n \bar{k}_i \cdot \sum_{j=1}^m p_{i,j} \cdot \theta^j \quad (6)$$

где \bar{k}_i — стадии метода Розенброка, $p_{i,j}$ — параметры задаваемые методом, $\theta = (t - t_n) / (t_{n+1} - t_n) = 0..1$.

Чаще всего бывает, что порядок (6) меньше чем задаваемый основной формулой. Так же формула (6) совпадает с основной при $\theta = 1$.

Итак, условия порядка записываются относительно параметра θ . Допустим, у нас должны выполняться четыре условия порядка (3-й порядок для автономных систем). Тогда необходимо выполнения $4 \cdot m$ условий порядка. Если параметров в (6) больше, то ещё n условий должно выполняться, для того, чтобы (6) совпала с основной формулой при $\theta = 1$. Если параметров больше чем $(4 \cdot m + n)$, то можно воспользоваться методом Ньютона с регуляризацией СЛАУ.

Для методов с комплексными коэффициентами решение ищется в виде:

$$y(\theta) = \sum_{i=1}^n \operatorname{Re} \left(\bar{k}_i \cdot \sum_{j=1}^m p_{i,j} \cdot \theta^j \right) \quad (7)$$

где \bar{k}_i — комплексные стадии, $p_{i,j}$ — комплексные параметры, $\theta = 0..1$.

2. Примеры уточнения методов Розенброка

Для взятых из литературы методов было проведено уточнение коэффициентов до точности 30 и более значащих цифр. Так же для них приводятся коэффициенты непрерывных методов основанных на стадиях данных методов.

2.1. (4,2)-метод

Данный метод взят из [1]. Это L-устойчивый метод четвертого порядка. Метод задается следующими формулами:

$$y_{n+1} = y_n + \sum_{i=1}^m p_i \cdot k_i, \quad D_n = E - a \cdot h \cdot f'_n$$

$$D_n \cdot k_1 = h \cdot f(y_n), \quad D_n \cdot k_2 = k_1$$

$$D_n \cdot k_3 = h \cdot f(y_n + \beta_{31} \cdot k_1 + \beta_{32} \cdot k_2) + \alpha_{32} \cdot k_2$$

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

Уточненные параметры даны в таблице 1. Там так же приводится непрерывный метод, основанный на формуле (6).

Таблица 1.

Параметры (4,2)-метода
$a=0.5728160624821348554080013849767683409315$
$p1=1.278369390124472506000782151942886989045$
$p2=-1.007386809804384747838093468437643128000$
$p3=0.9265539109395042110093604870343782187148$
$p4=-0.3339613183469116184167678944417856261224$
$\beta31=1.009004690299215025588082799604327399592$
$\beta32=-0.2590046902992150255880827996043273995917$
$\alpha32=-0.4955220641657818341715530456206093602956$
$\alpha42=-1.287776482339217217685184223893417502206$
Коэффициенты для непрерывного метода 3-го порядка
$p11=2.807940918382931280268331126232341095630$
$p12=-2.283352377926245217075188710786266412373$
$p13=0.7537808496677864428076397364968123057875$
$p21=-3.136230519398356699219158918564434846756$
$p22=4.602228694452115124287372736689335424252$
$p23=-2.473384984858143172906307286562543705497$
$p31=1.031459744165087582629789670058674398639$
$p32=-1.800682298774149742460551733666582983217$
$p33=1.695776465548566370840122550642286803293$
$p41=-1.031459744165087582629789670058674398639$
$p42=1.800682298774149742460551733666582983217$
$p43=-1.103183872955973778247529958049694210700$

2.2. CROS2_1

Данный метод взят из [3]. Это двухстадийный L1-устойчивый метод с комплексными коэффициентами 4-го порядка. Метод задается следующими формулами:

$$y_{n+1} = y_n + \text{Re}(b_1 \cdot k_1 + b_2 \cdot k_2)$$

$$[E - \tau \cdot \alpha_1 \cdot f_y(y_n)] \cdot k_1 = \tau \cdot f(y_n)$$

$$[E - \tau \cdot \alpha_2 \cdot f_y(y_n + \text{Re}(a \cdot k_1))] \cdot k_2 = \tau \cdot f(y_n + \text{Re}(c \cdot k_1))$$

Уточненные параметры даны в таблице 2. Там так же приводится непрерывный метод, основанный на формуле (7).

Таблица 2.

Параметры CROS2_1
$\alpha1= 0.09705048233513176225872820658157096372762+0.1441824711215367963888908188125269341956i$
$\alpha2= 0.1886638033791539520269860791327147505581+0.06177441689689081715947141717106675420030i$

$b_1 = 0.04833419895509438083025798424226646350104 - 0.3205959705202491275579925812180370610420i$
 $b_2 = 0.9516658010449056191697420157577335364990 - 1.696774337833590152863879033397701636829i$
 $c = 0.1730887968652108268236509028841256629989 - 0.1694095699539015207491928002966606686529i$
 $a = 0.5359744564304914728722121662633122018534 - 0.9665922748484189311614005696233999104339i$

Коэффициенты для непрерывного метода 3-го порядка

$p_{11} = 3.619032671469712553690516593029702603963 - 0.2004225284051934625395768416824377885643i$
 $p_{12} = -8.414446230029770923182842133180291069678 - 0.6613406243058764343155995103891857483354i$
 $p_{13} = 4.843747757515152750322583524392854929215 + 0.5411671821908207692971837708535864758580i$
 $p_{21} = -2.619032671469712553690516593029702603963 - 9.183665759393742328007731132919548994127i$
 $p_{22} = 8.414446230029770923182842133180291069678 + 29.50534469034316892770669484128541508572i$
 $p_{23} = -4.843747757515152750322583524392854929215 - 22.01845326878301675256284274176356772842i$

2.3. CROS2_Optimal

Данный метод взят из [4]. Это двухстадийный метод с комплексными коэффициентами 4-го порядка. Он имеет оптимальное L – затухание как для функции устойчивости, так и для функции внутренней устойчивости. Метод задается следующими формулами:

$$y_{n+1} = y_n + \text{Re}(\omega_1 \cdot k_1 + \omega_2 \cdot k_2)$$

$$[E - \tau \cdot \alpha_1 \cdot f_y(y_n)] \cdot k_1 = \tau \cdot f(y_n)$$

$$[E - \tau \cdot \alpha_2 \cdot f_y(y_n)] \cdot k_2 = \tau \cdot f(y_n + \text{Re}(\gamma_{21} \cdot k_1)) + \tau \cdot f_y(y_n + \text{Re}(\delta_{21} \cdot k_1)) \cdot \pi_{21} \cdot k_1$$

Уточненные параметры даны в таблице 3. Там так же приводится непрерывный метод, основанный на формуле (7).

Таблица 3.

Параметры CROS2_Optimal

$\alpha_1 = 0.4573733434972975729886712378597810560027 - 0.2351004879985426732069962332538443381790i$
 $\alpha_2 = 0.04262665650270242701132876214021894399728 - 0.3946329531721133798109289632446957436482i$
 $\gamma_{21} = 0.9147466869945951459773424757195621120054 + 0.6546908396281089262421198339615985775839i$
 $\delta_{21} = 0.7451903446015528669012176569684448709433 + 3.042955553310975129001697189423158388626i$
 $\pi_{21} = -0.2317220370463119616442110173716549151421 + 0.07081125027258135212790263920701540770149i$
 $\omega_1 = 0.2784065608064581775941260630978839609622 - 0.9196225438536243933605777235086716993250i$
 $\omega_2 = 0.7215934391935418224058739369021160390378 + 0.2014773772275646399542954916451697899548i$

Коэффициенты для непрерывного метода 3-го порядка

$p_{11} = 1.662352335632077153904266284657282585433 + 3.058378731739278701194655437881970784617i$
 $p_{12} = -1.145501425793116439658894990826412127076 - 6.527071235918501156496073309360094210372i$
 $p_{13} = -0.2384443490325025366512452307329864973956 + 2.549069960325598061940840147969451726429i$
 $p_{21} = -0.6623523356320771539042662846572825854333 - 3.084120463545428936697275176270423611394i$
 $p_{22} = 1.145501425793116439658894990826412127076 + 5.333814343596499665083881944741250473414i$
 $p_{23} = 0.2384443490325025366512452307329864973956 - 2.048216502823506088432311276825657072065i$

2.4. MyRosenbrock

Это авторский метод. Он основан методах Розенброка из [9]. S-стадийный метод Розенброка задается формулами:

$$k_i = h \cdot f\left(y_n + \sum_{j=1}^{i-1} \alpha_{ij} \cdot k_j\right) + h \cdot J \cdot \sum_{j=1}^i \gamma_{ij} \cdot k_j, \quad i = 1..s$$

$$y_{n+1} = y_n + \sum_{j=1}^s b_j \cdot k_j$$
(8)

где J — матрица Якоби.

Данный метод можно распространить и на неавтономные задачи с теми же коэффициентами (Подробности см. в [9]).

Метод 4-х стадийный, имеет 4-й порядок и он жестко точный. При этом:

$$\gamma_{11} = \gamma_{22} = \gamma_{33} = \gamma_{44} = \gamma$$

Уточненные параметры даны в таблице 4. Там так же приводится непрерывный метод, основанный на формуле (6).

Таблица 4.

Параметры метода MyRosenbrock
$\gamma=0.2204284102592123180415434126292381629422$
$\alpha_{21}=-0.5205432483291658413100814077709160752376$
$\alpha_{31}=0.6075130779487283175846362337845170732959$
$\alpha_{32}=0.1009846139765253545887270473713338894896$
$\alpha_{41}=0.1641108001779867014074129836895915640195$
$\alpha_{42}=0.1293432375977119743143904143048741327283$
$\alpha_{43}=0.7065459622243013242781966020055343032522$
$\gamma_{21}=1.041916673708566646332986903249560906166$
$\gamma_{31}=-0.01906306867082466862784767220621635695740$
$\gamma_{32}=-0.4883505673094258925645673717663805516965$
$\gamma_{41}=0.3208618397035608796737063652926844741367$
$\gamma_{42}=0.02205957636947700893965888505141667291981$
$\gamma_{43}=-0.5633498263322502066549086629733393099986$
$b_1=0.4849726398815475810811193489822760381562$
$b_2=0.1514028139671889832540492993562908056481$
$b_3=0.1431961358920511176232879390321949932536$
$b_4=0.2204284102592123180415434126292381629422$
Коэффициенты для непрерывного метода 3-го порядка
$p_{11}=1.614220168082677875270156338410046476717$
$p_{12}=-1.773522416520713007296954629873264838965$
$p_{13}=0.6442748883195827131079176404454944004042$
$p_{21}=-0.7730850179266677645012163439058016182438$
$p_{22}=2.000378477754902478764580585880475653431$
$p_{23}=-1.075890645861045731009314942618383229539$

p31=-0.1016288971935796221654533100265442181350
p32=0.6328462020633125972007704371496734160304
p33=-0.3880211689776818574120291880909342046418
p41=0.2604937470375695113965133155222993596616
p42=-0.8597022632975020686683963931568842304964
p43=0.8196369265191448753134264902638230337770

2.5. RODAS

Это метод Розенброка из [9], основанный на тех же формулах (8), что и предыдущий. Он 6-ти стадийный, вложенный, жестко точный.

Уточненные параметры даны в таблице 5. Там так же приводится непрерывный метод, основанный на формуле (6). При этом:

$$\gamma_{ii} = \gamma, \gamma_{ij} = \beta_{ij} - \alpha_{ij},$$

$$\hat{b}_i = \beta_{5i}, b_5 = \gamma \text{ — вложенный метод 3-го порядка,}$$

$$b_i = \beta_{6i}, b_6 = \gamma \text{ — основной метод 4-го порядка.}$$

Построение непрерывного метода рассматривается в [5].

Таблица 5.

Параметры метода RODAS

$\gamma=0.25$
 $\alpha_{21}=0.75$
 $\alpha_{31}=0.08612040081415230810139697878760241798691$
 $\alpha_{32}=0.1238795991858476818986030212123975820131$
 $\alpha_{41}=0.7749345355073239835959759197344993729071$
 $\alpha_{42}=0.1492651549508681389613302285878078259563$
 $\alpha_{43}=-0.2941996904581921225573061483223071988633$
 $\alpha_{51}=5.308746682646141097685244508085578815512$
 $\alpha_{52}=1.330892140037268841059079254823776347387$
 $\alpha_{53}=-5.374137811655560533686603914331557047750$
 $\alpha_{54}=-0.2655010110278494050577198485777981151496$
 $\alpha_{61}=-1.764437648774481929301295441577120895355$
 $\alpha_{62}=-0.4747565572063022542415003747471682133343$
 $\alpha_{63}=2.369691846915801042868069955427380559080$
 $\alpha_{64}=0.6195023590649831287129733353848142171276$
 $\alpha_{65}=0.2500000000000000119617525255120943324826$
 $\beta_{21}=0.0$
 $\beta_{31}=-0.04939199999999976379720604242479557432309$
 $\beta_{32}=-0.0141120000000002362027939575752044256769$
 $\beta_{41}=-0.4820494693877553295338356343641135140289$
 $\beta_{42}=-0.1008795555555553188031270166574940582696$

$\beta_{43}=0.9267290249433106483369626510216075722987$
 $\beta_{51}=-1.764437648774484950984492409473177979061$
 $\beta_{52}=-0.474756557206303164236822915996783772739$
 $\beta_{53}=2.369691846915804789575079569305906277052$
 $\beta_{54}=0.6195023590649833256462357561640554747468$
 $\beta_{61}=-0.080368370789111680412472246275789336523$
 $\beta_{62}=-0.0564906135924470382928821126158819718130$
 $\beta_{63}=0.488285630042796820218631292635299492216$
 $\beta_{64}=0.5057162114816190413438659233992289589771$
 $\beta_{65}=-0.1071428571428571428571428571428571428569$

Коэффициенты для непрерывного метода 3-го порядка

$p_{11}=0.4195946799873075119058720335388621401299$
 $p_{12}=-0.7335246808281686702714286385831914932339$
 $p_{13}=0.2677412801216726048848534454489047125024$
 $p_{14}=-0.03417965006992312693176908668036469592144$
 $p_{21}=0.2547354250795645221587858725988736232911$
 $p_{22}=-0.5928167839085814444337143623649121710159$
 $p_{23}=0.2900798161885619002781701285533800486149$
 $p_{24}=-0.008489070951992016296123751403223472703150$
 $p_{31}=1.051349424207045854163844526228495743638$
 $p_{32}=-1.025245760555795036484891571562738176784$
 $p_{33}=0.4239971915887983504197795780790293924265$
 $p_{34}=0.03818477480274765211989875989051253293510$
 $p_{41}=-0.8887851816599769967853215338953422550436$
 $p_{42}=3.271289083573963740495968694592938222440$
 $p_{43}=-1.879097079219423171177154225473452234601$
 $p_{44}=0.002309388787055468810372988175085226182488$
 $p_{51}=-0.0007361278248275885787333063783016651992557$
 $p_{52}=-0.5586980766585664375101099181839054056919$
 $p_{53}=0.2721840821423380149400326795532119142342$
 $p_{54}=0.1801072651981988682916676878661380138000$
 $p_{61}=0.1638417802108866971355524079074124131835$
 $p_{62}=-0.3610037816228521517958242038981909757140$
 $p_{63}=0.6250947091780523006543183938389261668234$
 $p_{64}=-0.1779327077660868459940465978481476042930$

2.6. RODAS5

Это метод Розенброка из [9], основанный на тех же формулах (8), что и предыдущий. Он 8-ми стадийный, вложенный, жестко точный. Построение данного метода можно найти в [7].

Уточненные параметры даны в таблице 6. При этом:

$$\gamma_{ii} = \gamma, \gamma_{ij} = \beta_{ij} - \alpha_{ij},$$

$$\hat{b}_i = \beta_{7i}, b_7 = \gamma \text{ — вложенный метод 4-го порядка,}$$

$$b_i = \beta_{8i}, b_8 = \gamma \text{ — основной метод 5-го порядка.}$$

Рабочий непрерывный метод не получилось построить, так как коэффициенты оказались слишком большими (порядка 10^{32}).

Таблица 6.

Параметры метода RODAS5	
$\gamma=0.19000000000000000000000000000000$	
$\alpha_{21}=0.38000000000000000000000000000000$	$\beta_{21}=-0.00384213367066316653924755891158$
$\alpha_{31}=0.18998096323048065165930209720647$	$\beta_{31}=-0.06587508126936369036787904015459$
$\alpha_{32}=0.18621536159480624834069790279353$	$\beta_{32}=-0.06331442359835370063212095984541$
$\alpha_{41}=0.083489136955726187158204690432470$	$\beta_{41}=0.53487908306342626417680465966545$
$\alpha_{42}=0.69567870317296510697170912192462$	$\beta_{42}=-0.47157078286588050578677711751138$
$\alpha_{43}=-0.27150977576330429412991381235709$	$\beta_{43}=0.44331311780172184160997245784594$
$\alpha_{51}=0.019749954685320738490462476406881$	$\beta_{51}=0.73026015628341180828986744008974$
$\alpha_{52}=1.1527680537682619629021698267018$	$\beta_{52}=-0.2532267377201464585976956105773$
$\alpha_{53}=-0.58172139560641273372994044079831$	$\beta_{53}=0.37740086754994634597527595020071$
$\alpha_{54}=-0.11569326366863669068176690949660$	$\beta_{54}=-0.098347581382224195667447779713122$
$\alpha_{61}=0.83441493842466974700123546967678$	$\beta_{61}=3.4243826682568670302497950472480$
$\alpha_{62}=1.4341150127667629897852468982053$	$\beta_{62}=-9.0682947038850299925361899111267$
$\alpha_{63}=-2.7463052995987880016901240308337$	$\beta_{63}=3.6910912705994601310908139407904$
$\alpha_{64}=4.63635670098841300000000000000000$	$\beta_{64}=5.9760755264645538704774440136053$
$\alpha_{65}=-3.1585813525810578672248231125632$	$\beta_{65}=-3.2132547614358500001592354994989$
$\alpha_{71}=3.4243826682568668406794981276729$	$\beta_{71}=0.3754294878045746740337485322662$
$\alpha_{72}=-9.0682947038850302879842814456849$	$\beta_{72}=-0.781123376542690E-16$
$\alpha_{73}=3.6910912705994596427861075132406$	$\beta_{73}=-0.1985496379538313447723579795563$
$\alpha_{74}=5.9760755264645538521206406306885$	$\beta_{74}=1.3845456784046259182088969764613$
$\alpha_{75}=-3.2132547614358500195546952967799$	$\beta_{75}=-0.6985967635977040646521761136264$
$\alpha_{76}=0.18999999999999999999999999999999$	$\beta_{76}=-0.05282876465766510470577376127586$
$\alpha_{81}=0.37542948780457440998356143171437$	$\beta_{81}=0.37542948780457467127309153966693$
$\alpha_{82}=-0.46222173916448809079146613083026E-16$	$\beta_{82}=-0.71947341385872637662088537334586E-16$
$\alpha_{83}=-0.19854963795383124505629117057112$	$\beta_{83}=-0.19854963795383134239547860424177$
$\alpha_{84}=1.3845456784046262637970502755724$	$\beta_{84}=1.3845456784046259068336921183261$
$\alpha_{85}=-0.69859676359770365574433060778458$	$\beta_{85}=-0.69859676359770406041453472873944$
$\alpha_{86}=-0.052828764657665387985155187551153$	$\beta_{86}=-0.051105182319437447937208832186395$
$\alpha_{87}=0.18999999999999999999999999999999$	$\beta_{87}=-0.00172358233822765541222010695281$

3. Тестирование

В целях проведения тестирования была написана специальная программа. Программа позволяет получать экстраполяционное решение на различных сетках, решение на основе контроля погрешности, сравнивать решения, строить графики и фазовые портреты, а так же вычислять порядок и тестировать методы. В ней реализовано около 130 численных методов.

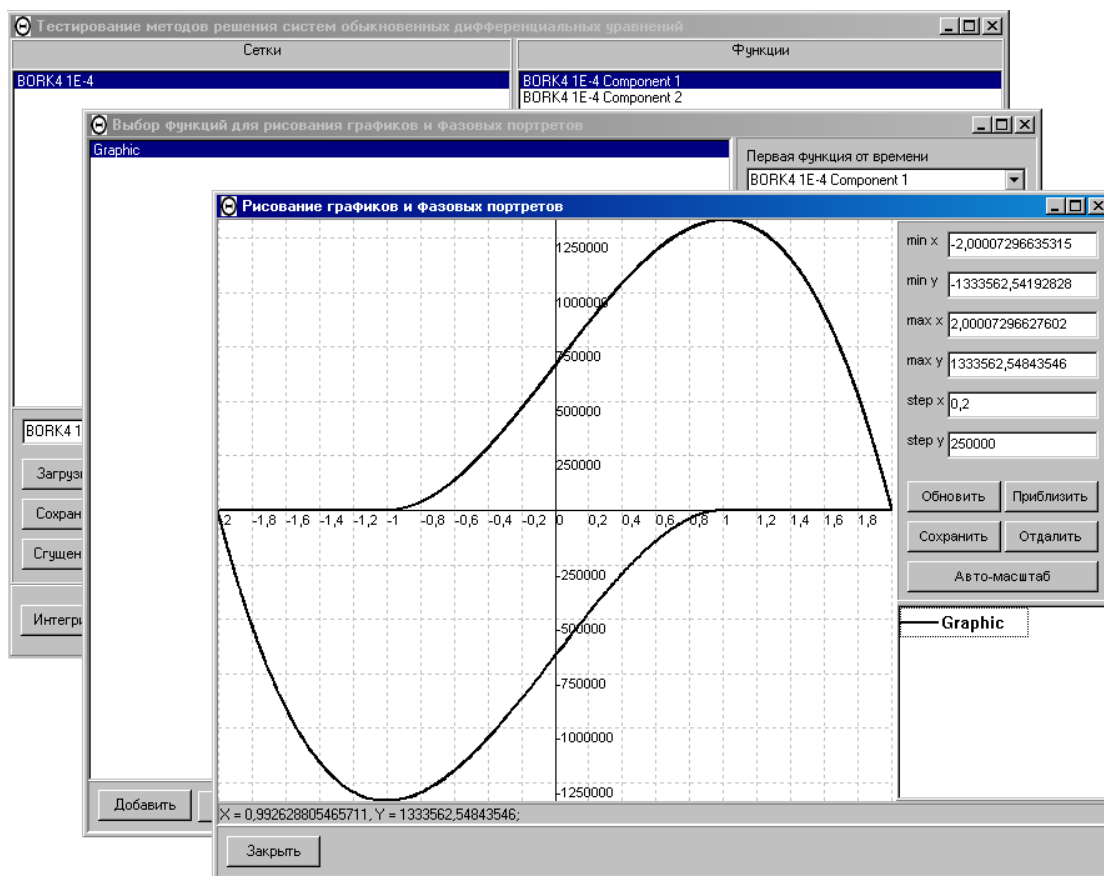


Рисунок 2. Программа для тестирования современных численных методов решения ОДУ

Все тесты проводились при повышенной вычислительной точности (38-39 цифр).

3.1. Выбранные методы.

Все тестируемые методы держат свой порядок на нежестких задачах для точности более 30 значащих десятичных цифр, так как коэффициенты всех методов были уточнены до 30 и более цифр. Для тестирования были выбраны следующие методы:

- Multirate (4,2) — непрерывный на основе 4-х стадийного с 2 вычислениями правой части метода Розенброка, порядок 4 (Новиков Е.А.[1]);
- BORK4 — L4-устойчивый обратный оптимальный неявный метод Рунге-Кутты из диссертации Пошивайло И.П. [2], порядок 4, с интерполяцией Эрмита 3-й степени по двум ближайшим точкам 4-го порядка;
- MRCROS2_1 — непрерывный на основе двухстадийного L1-устойчивого метода Лимонова А. Г. [3] с комплексными коэффициентами 4-го порядка;
- MRCROS2_Optimal — непрерывный на основе двухстадийного оптимально затухающего метода Ширкова П.Д. [4] из статьи 1992 года с комплексными коэффициентами, порядок 4(3);
- MRRos — авторский непрерывный жестко точный 4-х стадийный A-устойчивый метод Розенброка (Власов С.А.), порядок 4(3);

- MRRODAS — непрерывный жестко точный 6-ти стадийный вложенный А-устойчивый метод Розенброка, порядок 4(3) (V.Savcenco [5]);
- RIIA_B2 — линейно неявная схема LN-эквивалентная неявному методу RadauIIA (Семейство Б) 3-го порядка (Зубанов А.М. [6]), с интерполяцией Ньютона 2-й степени по трем ближайшим точкам 3-го порядка;
- GearAdapt — многошаговый метод Гира [13] на основе формул дифференцирования назад с разгоном от 1-го до 4-го порядка, с интерполяцией Эрмита 3-й степени по двум ближайшим точкам 4-го порядка;
- RODAS5 — жестко точный 8-ми стадийный вложенный А-устойчивый метод Розенброка, порядок 5(4) (Giovanna A. Di Marzo [7]), с интерполяцией Ньютона 4-й степени по пяти ближайшим точкам 5-го порядка;
- SkvorcovAdapt — явный многошаговый адаптивный метод с разгоном от 1-го до 5-го порядка (Скворцов Л.М. [14]), с интерполяцией Эрмита 5-й степени по трем ближайшим точкам 6-го порядка;
- ESDIRK86 — классический метод, основанный на 9-ти стадийном однократно диагонально-неявном методе Рунге-Кутты из статьи Скворцова Л.М. [8] 6-го порядка и интерполяции Ньютона 5-й степени по шести ближайшим точкам 6-го порядка или интерполяции Эрмита 5-й степени по трем ближайшим точкам 6-го порядка;
- Lobatto IIIС8 — классический метод, основанный на полностью неявном 5-ти стадийном методе Рунге-Кутты Lobatto IIIС8 [9] и интерполяции Ньютона 7-й степени по восьми ближайшим точкам 8-го порядка или интерполяции Эрмита 7-й степени по четырём ближайшим точкам 8-го порядка;
- RADAU IIA15 — классический метод, основанный на полностью неявном 8-ми стадийном методе Рунге-Кутты RADAU IIA15 [9] и интерполяции Эрмита 15-й степени по восьми ближайшим точкам 16-го порядка;
- Spline7 — авторский непрерывный коллокационный А-устойчивый метод (Власов С.А. [10]), сплайн 7-й степени, который имеет 8-й порядок;

3.2. Стратегия выбора шага

Размер шага выбирается следующим образом:

$$h_{i,0} = h_{i-1} \cdot p \sqrt{\frac{Tol}{|err_{i-1}|}} \cdot kof ,$$

где h_{i-1} — предыдущий шаг, $|err_{i-1}|$ — абсолютная погрешность на предыдущем шаге, Tol — задаваемая погрешность, p — порядок метода, $kof \leq 0,9$ — некий коэффициент (в случае интерполяции решения он позволяет надлежащим образом дробить шаг для методов высокого порядка).

Далее оценивается текущая погрешность либо по правилу Рунге, либо способом, предусмотренным решаемым методом, и проверяется условие:

$$|err_{i,0}| < Tol .$$

Если оно не выполняется, то текущий шаг отбрасывается, и производятся вычисления для уменьшенного шага до тех пор, пока условие не выполнится, в соответствии с формулой:

$$h_{i,j} = h_{i,j-1} \cdot p \sqrt{\frac{Tol}{|err_{i,j}|}} \cdot kof ,$$

где j — номер попытки.

В случае, когда решение интерполируется и используется метод порядка больше седьмого, имеет смысл дополнительно дробить шаг на число равное количеству интерполируемых точек, с помощью коэффициента kof в формулах выше.

3.3. Задача Капса

Это автономная сверхжесткая (из-за параметра) тестовая задача [15], содержащая две компоненты решения и не имеющая запаздываний:

$$\begin{cases} \dot{x}_1 = -(\mu + 2) \cdot x_1 + \mu \cdot x_2^2 \\ \dot{x}_2 = x_1 - x_2 - x_2^2 \end{cases}, t \in (0, T]$$

где $\mu = 10^{12}$, $T = 1$.

Начальная задача Коши:

$$\begin{cases} x_1(0) = 1 \\ x_2(0) = 1 \end{cases}$$

Задача Капса имеет плавное решение $x_1(t) = \exp(-2t)$, $x_2(t) = \exp(-t)$, не зависящее от параметра жесткости μ (собственные значения якобиана при больших μ примерно равны $-\mu$, -1).

Несмотря на то, что задача Коши не имеет запаздываний, было принято решение интерполировать результирующие функции, получаемые в разностном виде, чтобы посмотреть точность интерполяции. Следует так же заметить, что матрица Якоби для методов Розенброка считается аналитически, то есть точно. В качестве оценки погрешности методов была выбрана норма С (для разностей приближенного решения и точного решения (для двух компонент) в точках на всем отрезке интегрирования (от 0 до 1) с постоянным интервалом (0,01)). Количество шагов у всех методов ограничено одним миллионом. Результаты тестирования показаны на рисунках 3, 4, 5, 6.

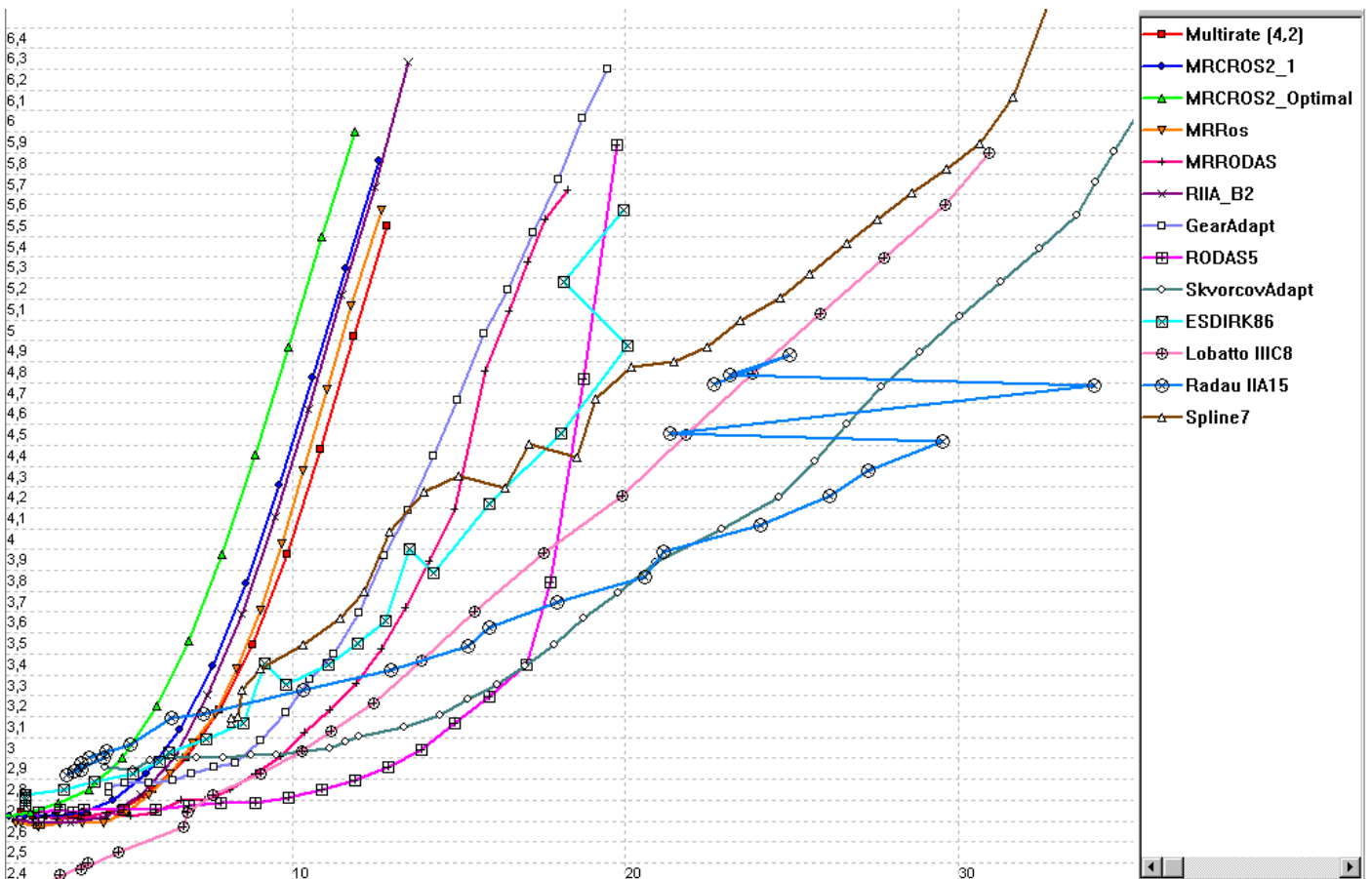


Рисунок 3. Достигнутая точность — трудоемкость.

Точность на рисунках отображает количество значащих цифр, а трудоемкость считается как $\log_{10}(time)$, где $time$ — время выполнения программы в миллисекундах.

BORK4 не показан на рисунках, так как он не справился с задачей (не смотря на то, что используется модифицированный метод Ньютона с дроблением шага). Причина может быть в плохой обусловленности СЛАУ, которую приходится решать на итерациях Ньютона. Данный метод работает на менее жесткой задаче с параметром $\mu = 10^6$. При текущей же жесткости работает BORK3 и BORK2 [2], но они имеют 3-й и 2-й порядок соответственно.

Неожиданный результат получился у RODAS5: до точности в 15 значащих цифр он имеет 5-й порядок, а после скатывается до первого, причем только по первой компоненте решения.

Свою работоспособность показал авторский непрерывный жестко точный 4-х стадийный А-устойчивый метод Розенброка 4-го порядка для разностного решения и 3-го порядка для интерполяционного.

По результатам тестирования все методы можно разбить на 3 класса точности: те, с помощью которых можно получить решение с инженерной точностью (5-6 значащих цифр), те, которые рассчитаны на получение решения высокой точности (10-12 значащих цифр), и те, которые могут считать со сверх высокой точностью (20-30 значащих цифр). Разбиение выглядит следующим образом:

1. Multirate (4,2), MRCROS2_1, MRCROS2_Optimal, MRRos, RIIA_B2;
2. MRRODAS, GearAdapt, RODAS5, ESDIRK86;
3. SkvorcovAdapt, Lobatto IIIС8, RADAU IIA15, Spline7.

В классе инженерной точности наилучшее соотношение ($\lg(\|Tol\|) - \lg(\|error\|)$) трудно выявить.

В классе высокой точности наилучшее соотношение ($\lg(\|Tol\|) - \lg(\|error\|)$) у RODAS5.

В классе сверх высокой точности наилучшее соотношение ($\lg(\|Tol\|) - \lg(\|error\|)$) у SkvorcovAdapt, он же и самый точный на данной задаче.

В [15] отмечается, что данная задача — это пример сингулярно возмущенной задачи Коши.

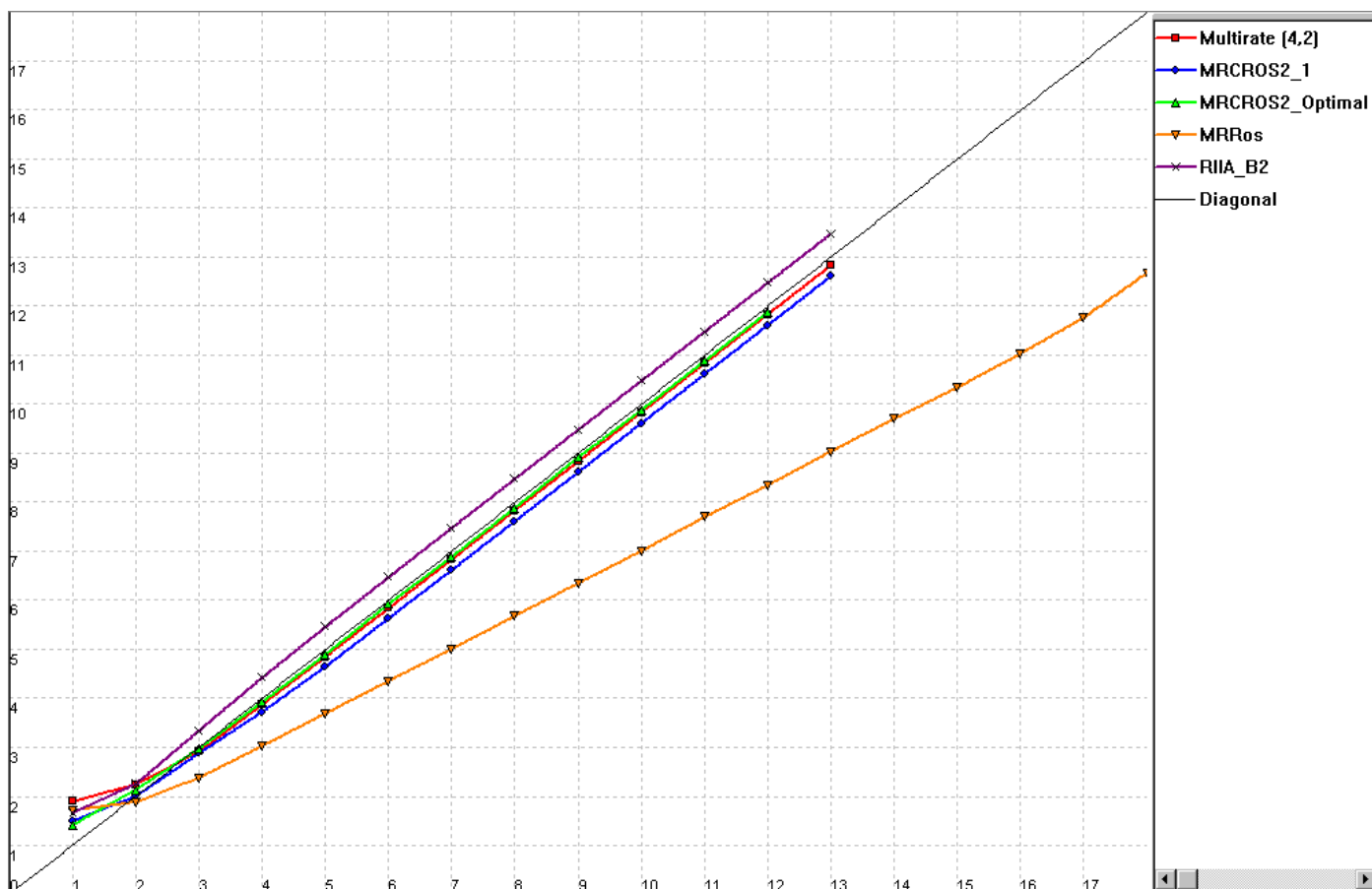


Рисунок 4. Задаваемая точность - достигаемая точность для класса инженерной точности.

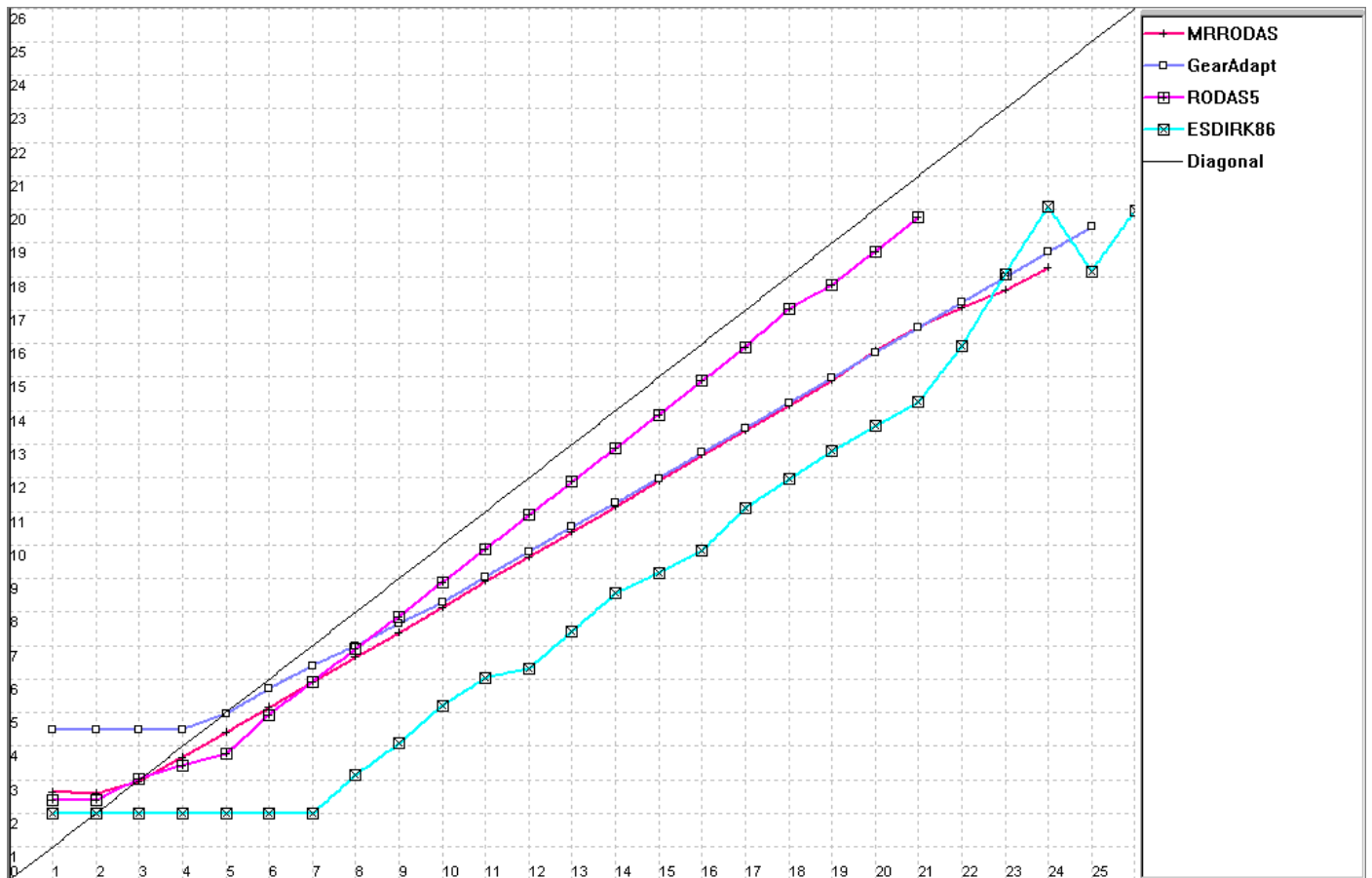


Рисунок 5. Задаваемая точность - достигаемая точность для класса высокой точности.

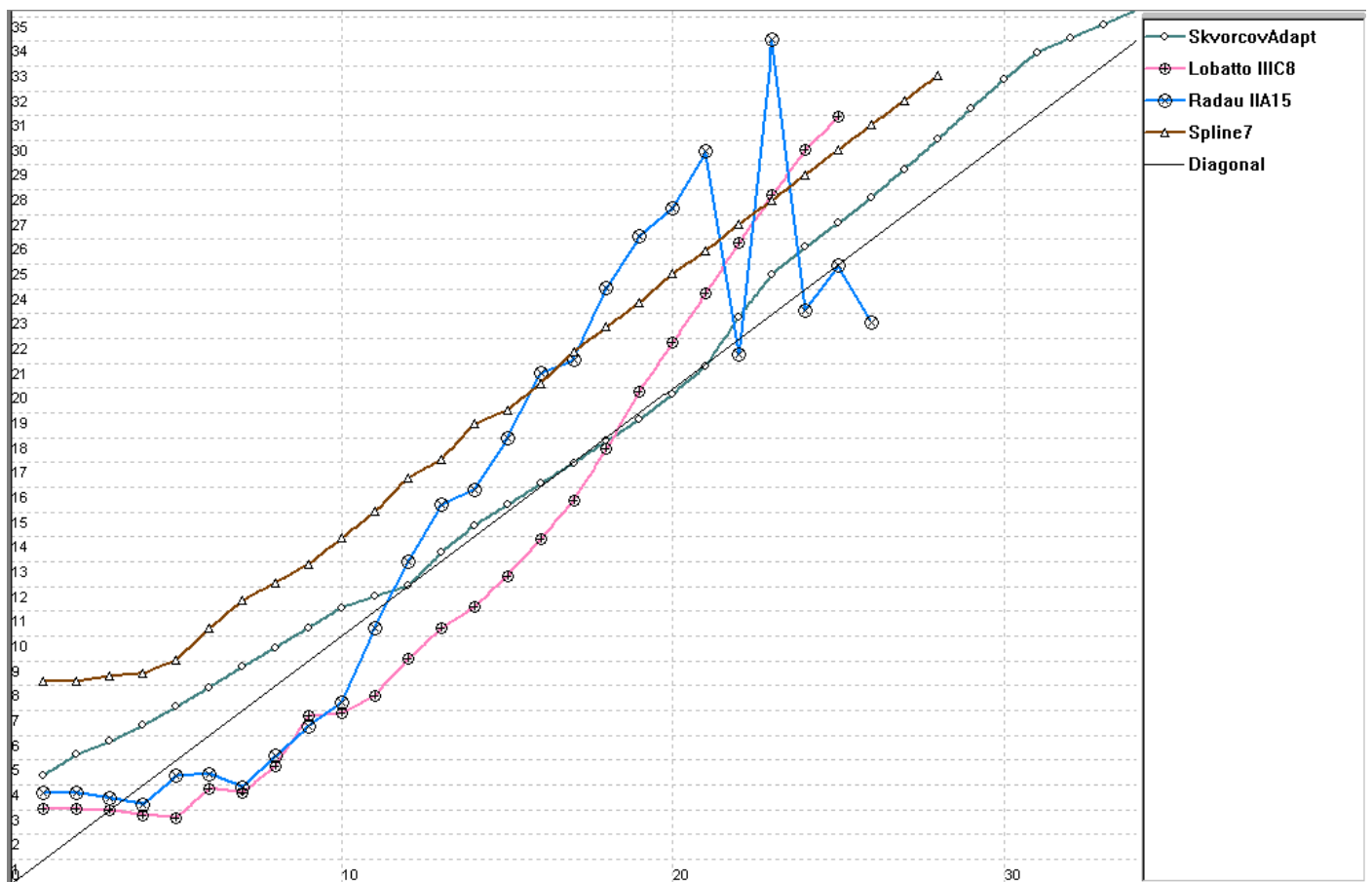


Рисунок 6. Задаваемая точность - достигаемая точность для класса сверх высокой точности.

3.3. Пример Крайсса

Это неавтономная сверхжесткая (из-за параметра) тестовая задача [15], содержащая две компоненты решения и не имеющая запаздываний:

$$\dot{y}(t) = E(t) \cdot \begin{bmatrix} -\varepsilon^{-1} & 0 \\ 0 & -1 \end{bmatrix} \cdot E^{-1}(t) \cdot y(t), \quad 0 < \varepsilon \leq 1, \quad 0 \leq t \leq 3,$$

где $\varepsilon = 10^{-12}$, $E(t) = \begin{bmatrix} \cos(t) & -\sin(t) \\ \sin(t) & \cos(t) \end{bmatrix}$.

Начальная задача Коши:

$$\begin{cases} y_1(0) = 1 \\ y_2(0) = 3 \end{cases}$$

Имеет плавное решение, которое почти не зависит от параметра, отвечающего за жесткость.

Эталонное решение было построено явным многошаговым адаптивным методом Скворцова с контролем погрешности $eps = 10^{-29}$. Внешний вид решения показан на рисунке 7.

Результирующие функции, получаемые в разностном виде, были интерполированы, чтобы посмотреть точность интерполяции. Матрица Якоби для методов Розенброка считается аналитически, то есть точно. Так как задача неавтономная, там, где это нужно, была введена дополнительная переменная для создания автономности. В качестве оценки погрешности методов была выбрана норма C (для разностей приближенного решения и полученного эталонного решения (для двух компонент) в точках на всем отрезке интегрирования (от 0 до 3) с постоянным интервалом (0,01)). Количество шагов у всех методов ограничено одним миллионом. Результаты тестирования показаны на рисунках 8, 9, 10, 11.

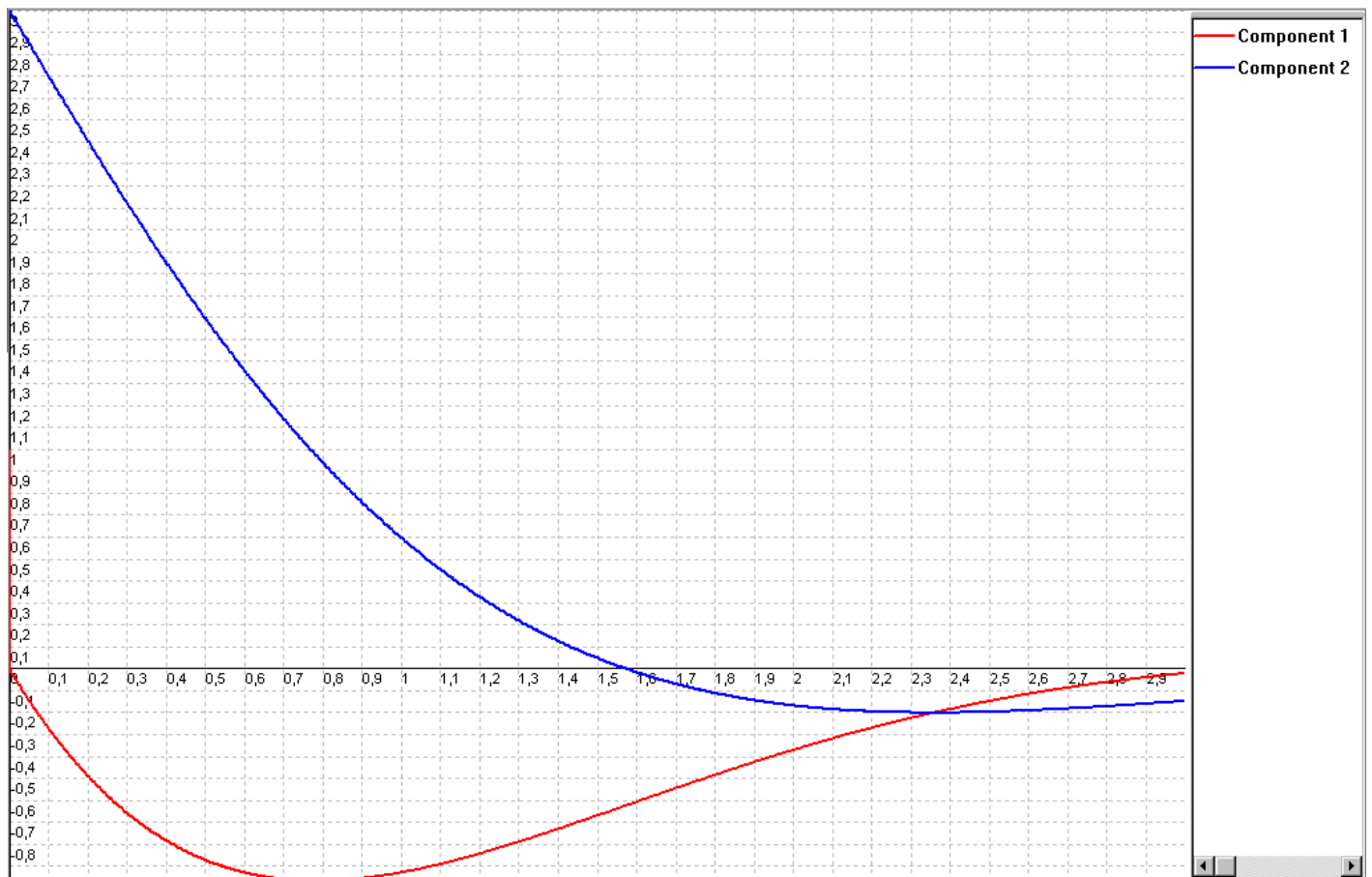


Рисунок 7. Эталонное решение примера Крайсса.

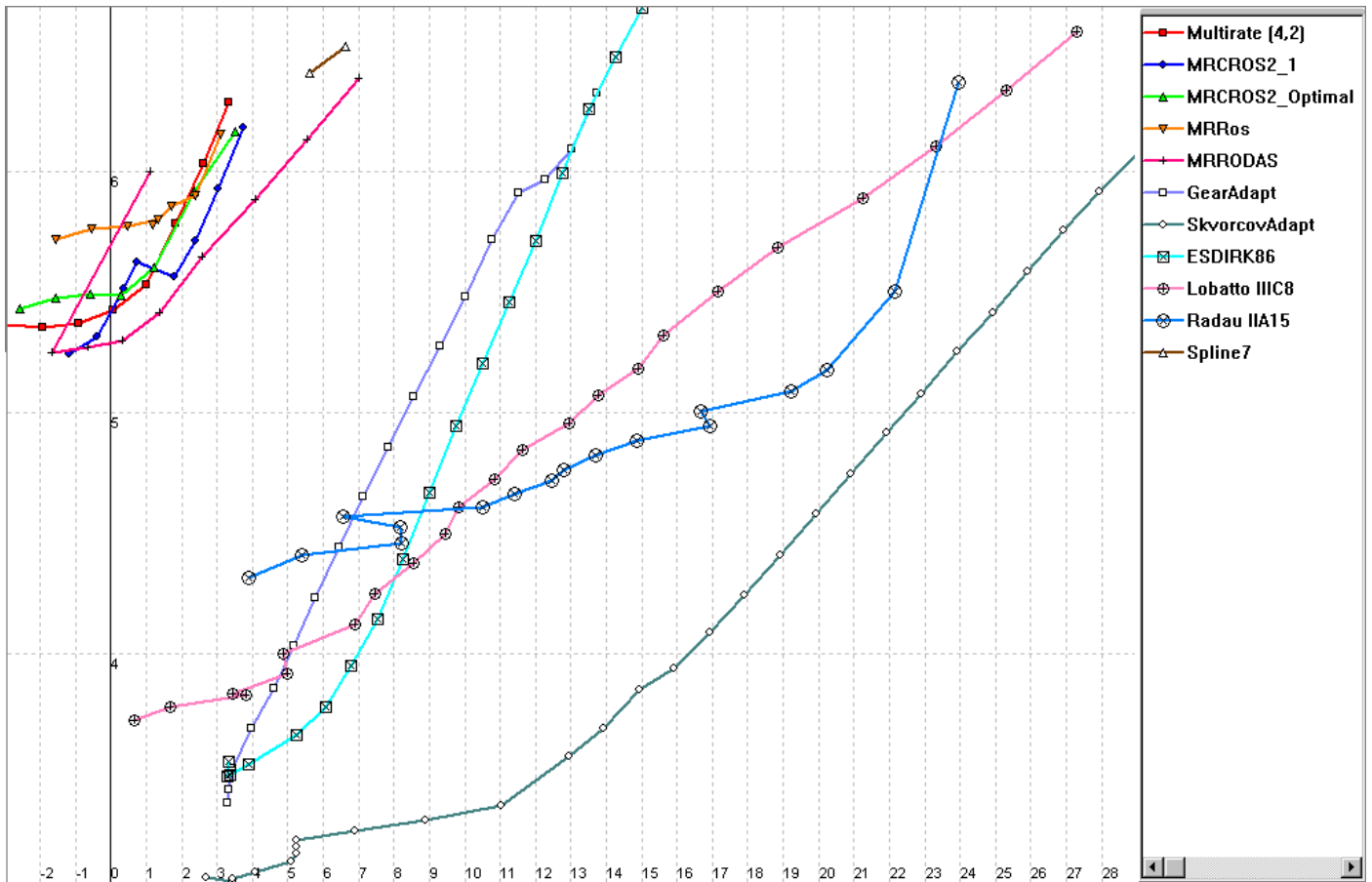


Рисунок 8. Достигнутая точность — трудоемкость.

BORK4, не показан на рисунках, так как он опять не справился с задачей. Но он работает на менее жесткой задаче с параметром $\varepsilon = 10^{-6}$.

RPA_B2, RODAS5 работают, но слишком медленно, поэтому они тоже не показаны на рисунке.

По результатам тестирования все методы можно разбить на 3 класса точности: те, с помощью которых можно получить решение с низкой точностью (2-3 значащие цифры), те, которые рассчитаны на получение решения высокой точности (10-12 значащих цифр), и те, которые могут считать со сверх высокой точностью (20-30 значащих цифр). Разбиение выглядит следующим образом:

1. Multirate (4,2), MRCROS2_1, MRCROS2_Optimal, MRRos, MRRODAS, Spline7.
2. GearAdapt и ESDIRK86.
3. SkvorcovAdapt, Lobatto III C8, RADAU IIA15.

Следует отметить, что все методы Розенброка работали на много лучше при меньшей жесткости с параметром $\varepsilon = 10^{-6}$.

В классе инженерной точности наилучшее соотношение ($\lg(\|Tol\|) - \lg(\|error\|)$) у MRRODAS.

В классе высокой точности наилучшее соотношение ($\lg(\|Tol\|) - \lg(\|error\|)$) у ESDIRK86.

В классе сверх высокой точности наилучшее соотношение ($\lg(\|Tol\|) - \lg(\|error\|)$) у SkvorcovAdapt, он же и самый точный на данной задаче.

Задача диссипативна в евклидовой норме, так как матрица $E(t)$ является ортогональной при всех t . Точнее, задача принадлежит к классу $\mathcal{F}_{\nu=-1}$, где $\nu=-1$ — постоянная в одностороннем условии Липшица.

Ван Вельдхейзен ввел класс задач \mathcal{P} , к которому можно отнести и данную задачу [15]. Эти задачи допускают одновременное вхождение в решение медленных и быстрых компонент, содержат малый параметр, позволяющий перейти к произвольно высокой степени жесткости, и имеют зависящие от времени собственные векторы. Вводится данный класс вместе с определением о D-устойчивости.

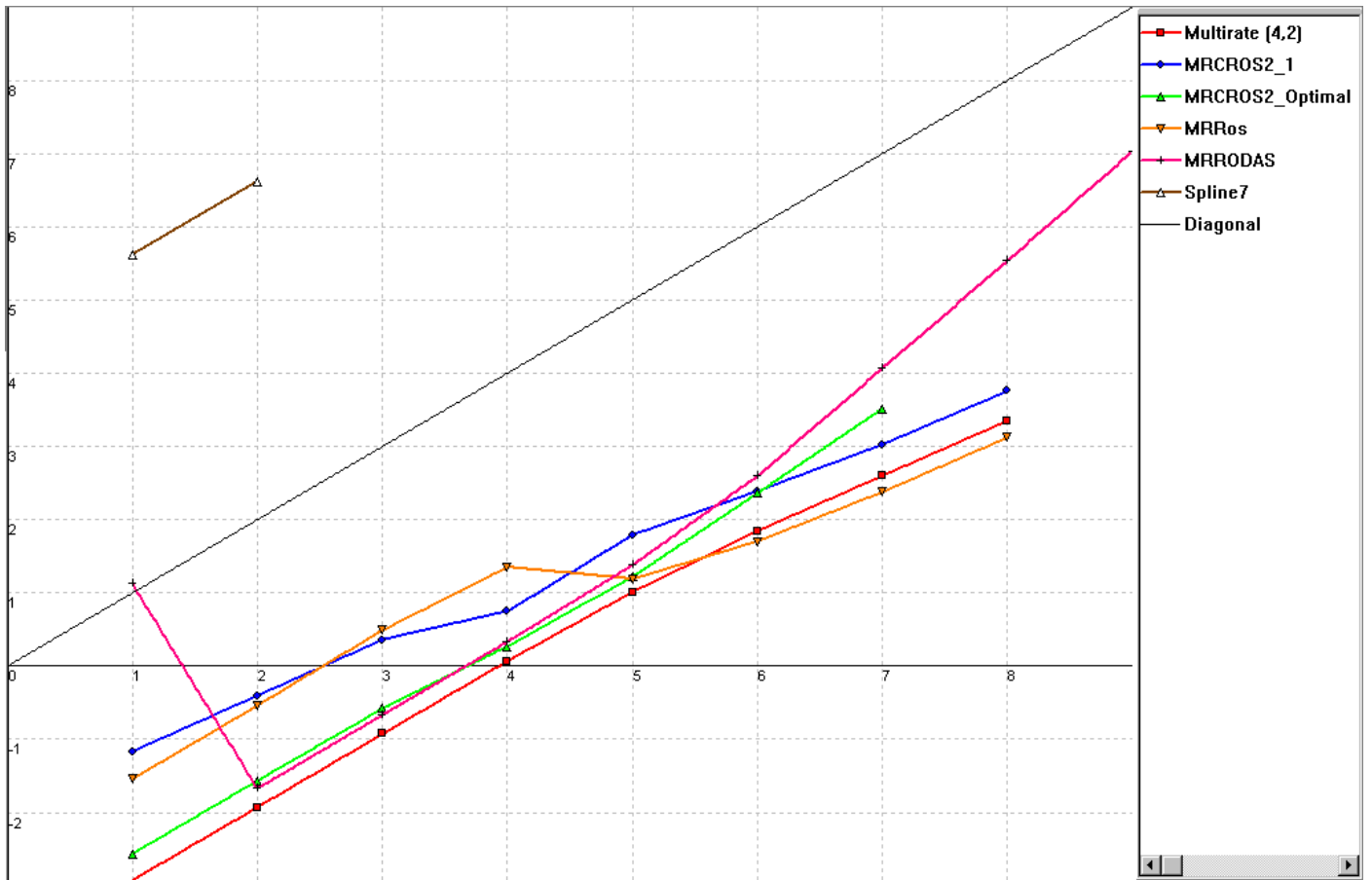


Рисунок 9. Задаваемая точность - достигаемая точность для класса низкой точности.

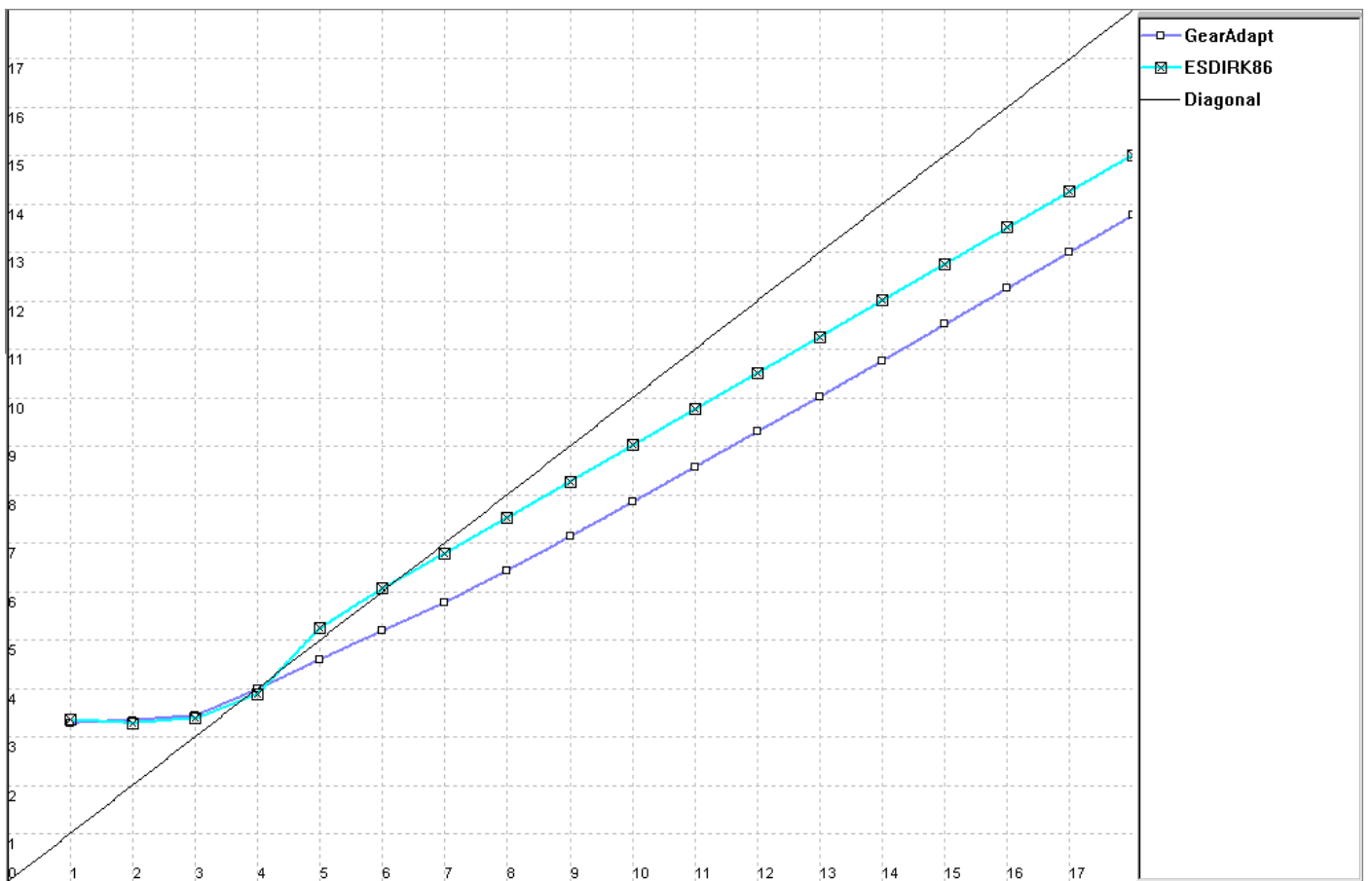


Рисунок 10. Задаваемая точность - достигаемая точность для класса высокой точности.

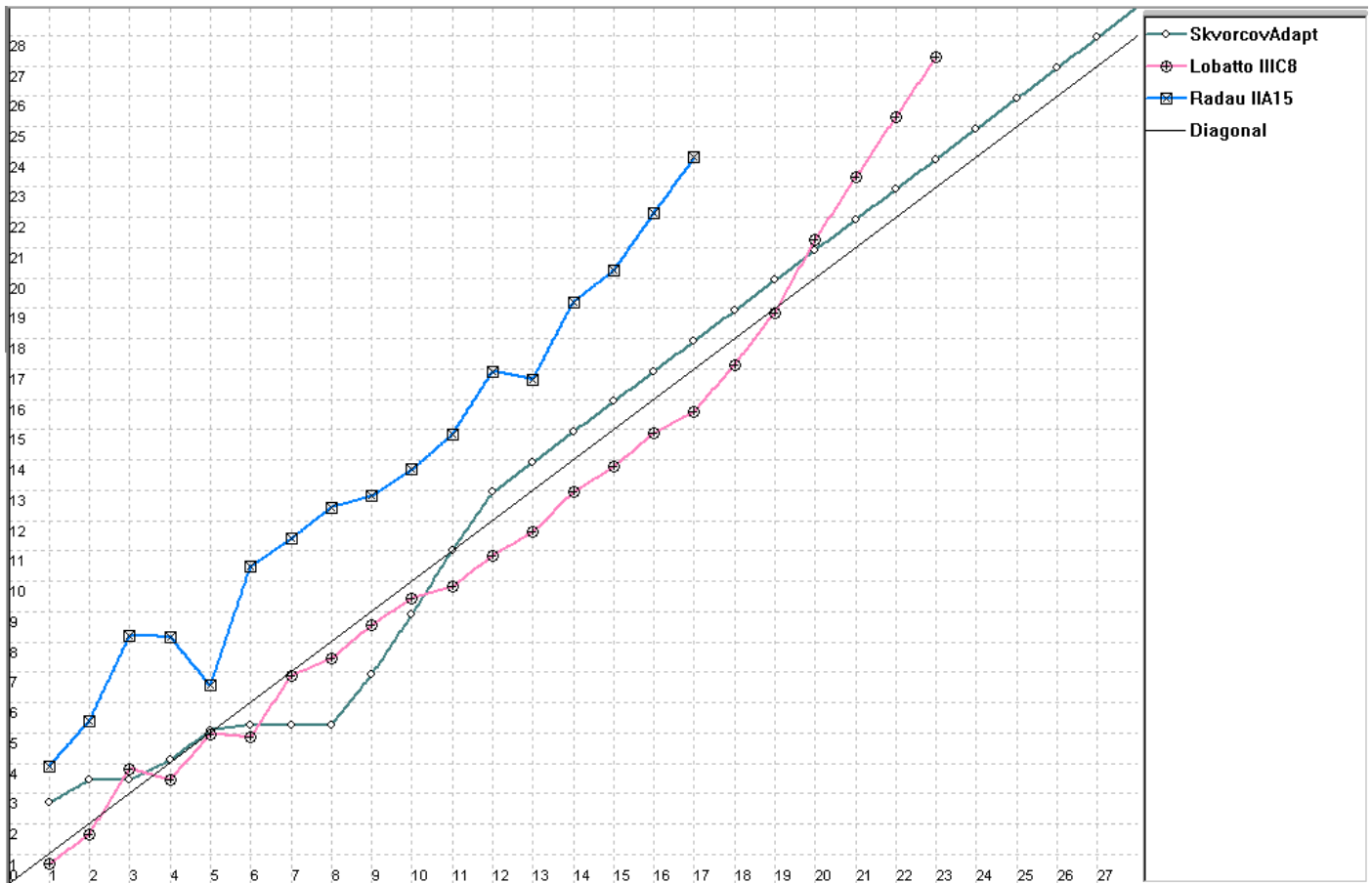


Рисунок 11. Задаваемая точность - достигаемая точность для класса сверх высокой точности.

3.4. Осциллятор ван дер Поля

Это автономная сверхжесткая (из-за параметра) тестовая задача, содержащая две компоненты решения и не имеющая запаздываний. Осциллятор с нелинейным затуханием подчиняется следующей системе уравнений:

$$\begin{cases} \dot{x} = y \\ \dot{y} = -a \cdot (y \cdot (x^2 - 1) + x) \end{cases}, t \in (0, T]$$

где $a = 10^{12}$, $T = 2$.

Начальная задача Коши:

$$\begin{cases} x(0) = 2 \\ y(0) = 0 \end{cases}$$

Решение было построено сплайн методом [10] 7-й степени, порядок 8, с контролем погрешности $eps = 10^{-10}$ (Основная сетка: 23 485 узлов).

Ещё два решения были получены на двух сетках, которые сгустили в 2 и в 4 раза, для оценки локальной погрешности.

Правило Рунге-Ричардсона позволяет численно оценить локальную погрешность и порядок метода:

$$p = \log_2 \left| \frac{u_1 - u_2}{u_2 - u_4} \right|,$$

$$error = \frac{u_1 - u_2}{1 - \frac{1}{2^p}},$$

$$y = u_1 - error,$$

где p — порядок, $error$ — погрешность, u_1, u_2, u_4 — решения на сгущающихся сетках, y — эталонное решение.

Результаты экстраполяции показаны на рисунках 12, 13, 14, 15.

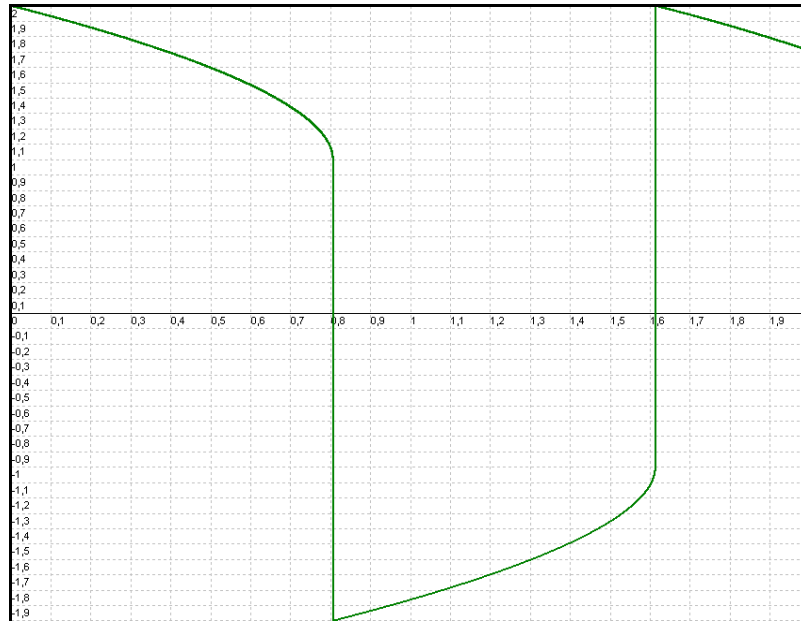


Рисунок 12. Эталонное решение (1-я компонента).

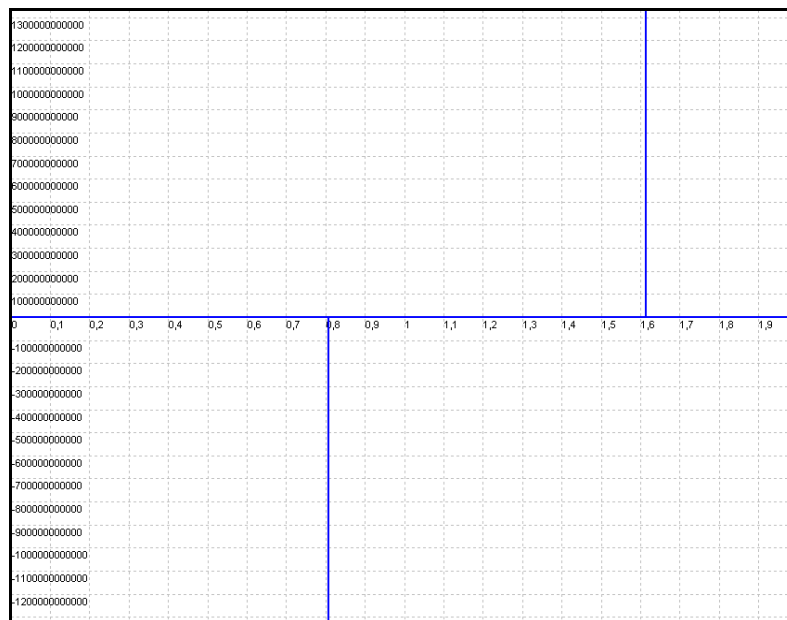


Рисунок 13. Эталонное решение (2-я компонента).

Результирующие функции, получаемые в разностном виде, были интерполированы, чтобы посмотреть точность интерполяции. Матрица Якоби для методов Розенброка считается аналитически, то есть точно. В качестве оценки погрешности методов была выбрана норма C (для разностей приближенного решения и полученного эталонного решения (для двух компонент) в точках на всем отрезке интегрирования (от 0 до 2) с постоянным интервалом (0,01)). Количество шагов у всех методов ограничено одним миллионом. Результаты тестирования показаны на рисунках 16, 17, 18.

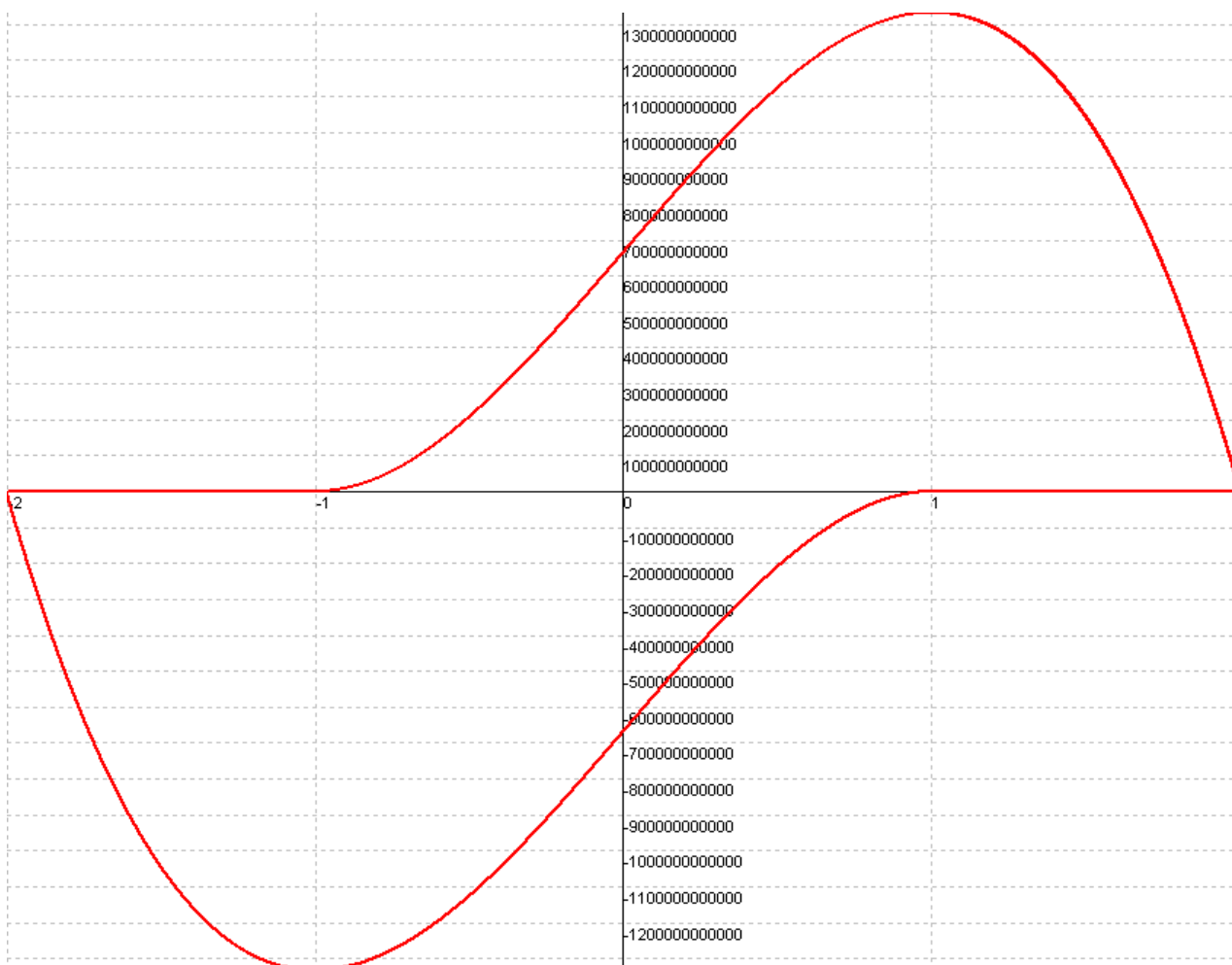


Рисунок 14. Эталонное решение (фазовый портрет).

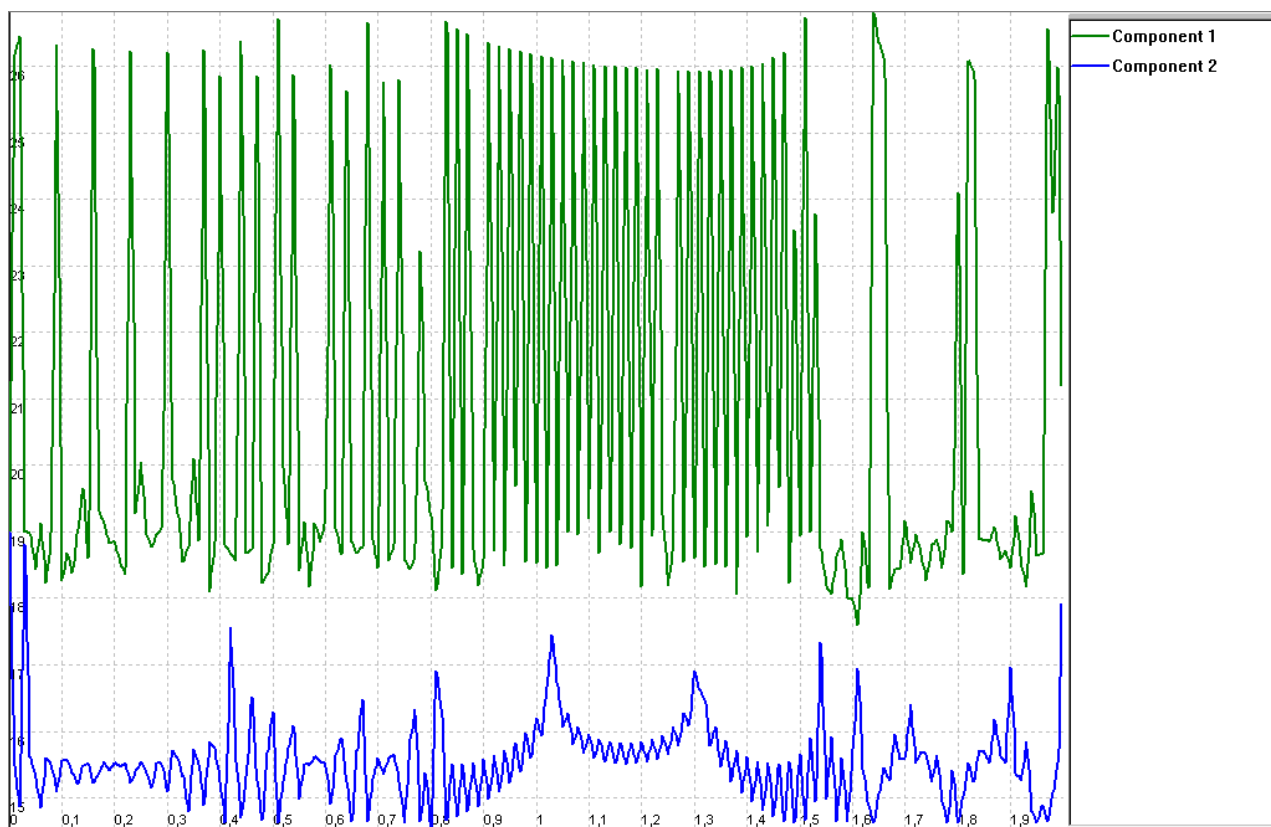


Рисунок 15. Локальная точность ($-\lg(|error|)$).

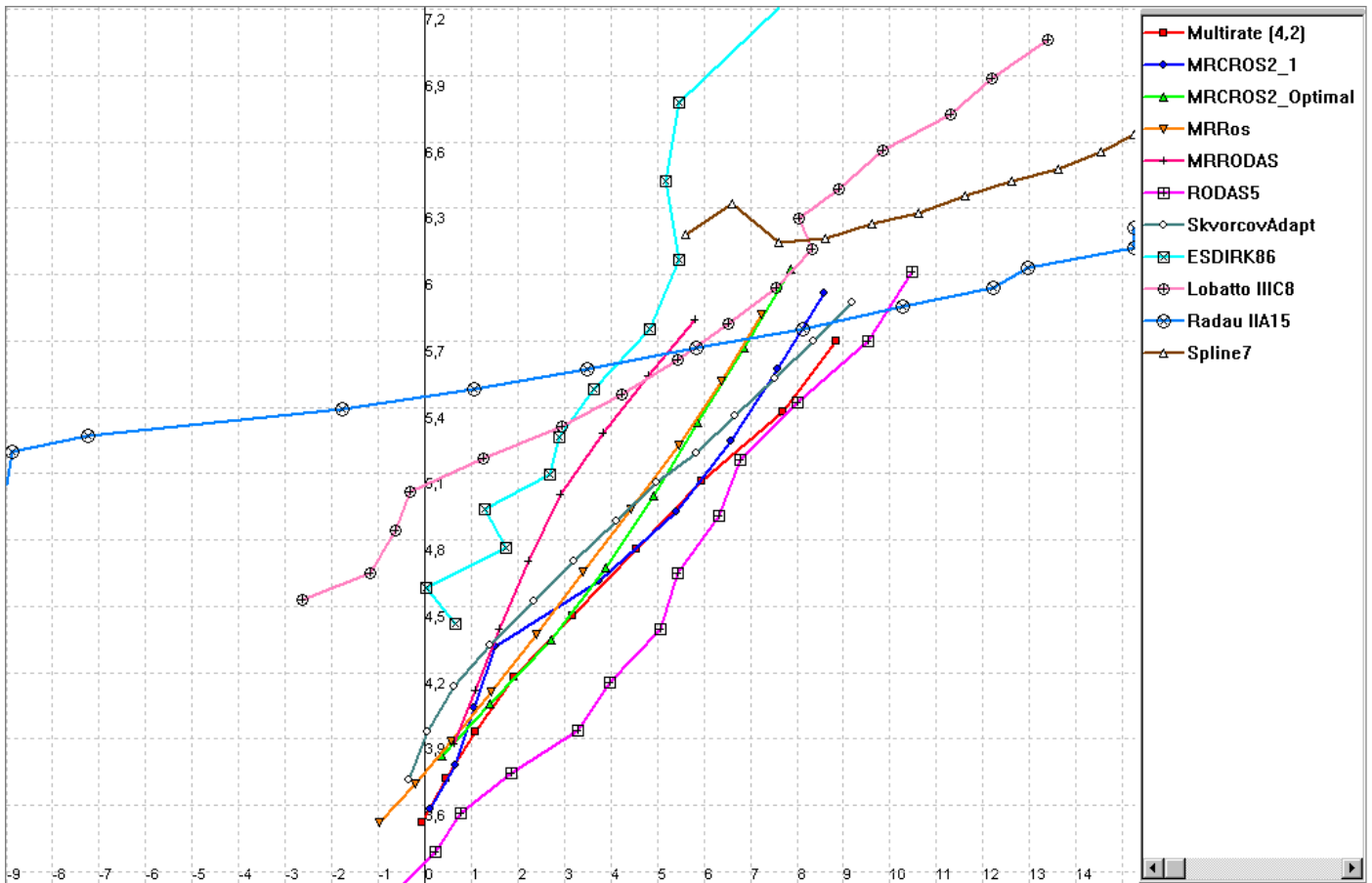


Рисунок 16. Достигнутая точность — трудоемкость.

BORK4 и RPA_B2 не показаны на рисунках, так как они не справились с задачей. Но они работают на менее жесткой задаче ван дер Поля с параметром $a = 10^6$.

Так как решение содержит быстро изменяющиеся компоненты, GearAdapt работает плохо, поэтому он не показан на рисунках.

По результатам тестирования все методы можно разбить на 2 класса точности: те, с помощью которых можно получить решение с инженерной точностью (5-6 значащих цифр), и те, которые рассчитаны на получение решения высокой точности (10-12 значащих цифр). Разбиение имеет следующий вид:

1. Multirate (4,2), MRCROS2_1, MRCROS2_Optimal, MRRos, MRRODAS, SkvorcovAdapt, ESDIRK86;
2. RODAS5, Lobatto III C8, RADAU IIA15, Spline7.

В первом классе наилучшее соотношение ($\lg(\|Tol\|) - \lg(\|error\|)$) у MRCROS2_Optimal.

Во втором классе наилучшее соотношение ($\lg(\|Tol\|) - \lg(\|error\|)$) у RODAS5.

Уравнение ван дер Поля применяется и в физике, и в биологии. Так, например, в биологии создана модель ФитцХью-Нагумо. Данное уравнение также было использовано в сейсмологии для моделирования геологических разломов.

Уравнение ван дер Поля используется в радиотехнике для описания не только триодного генератора, но и генератора на туннельном диоде или транзисторного генератора [16].

Один из ярких примеров: ван дер Поль (в соавторстве с ван дер Марком) обсуждает эффективность применения этого уравнения для описания кардиоритмов.

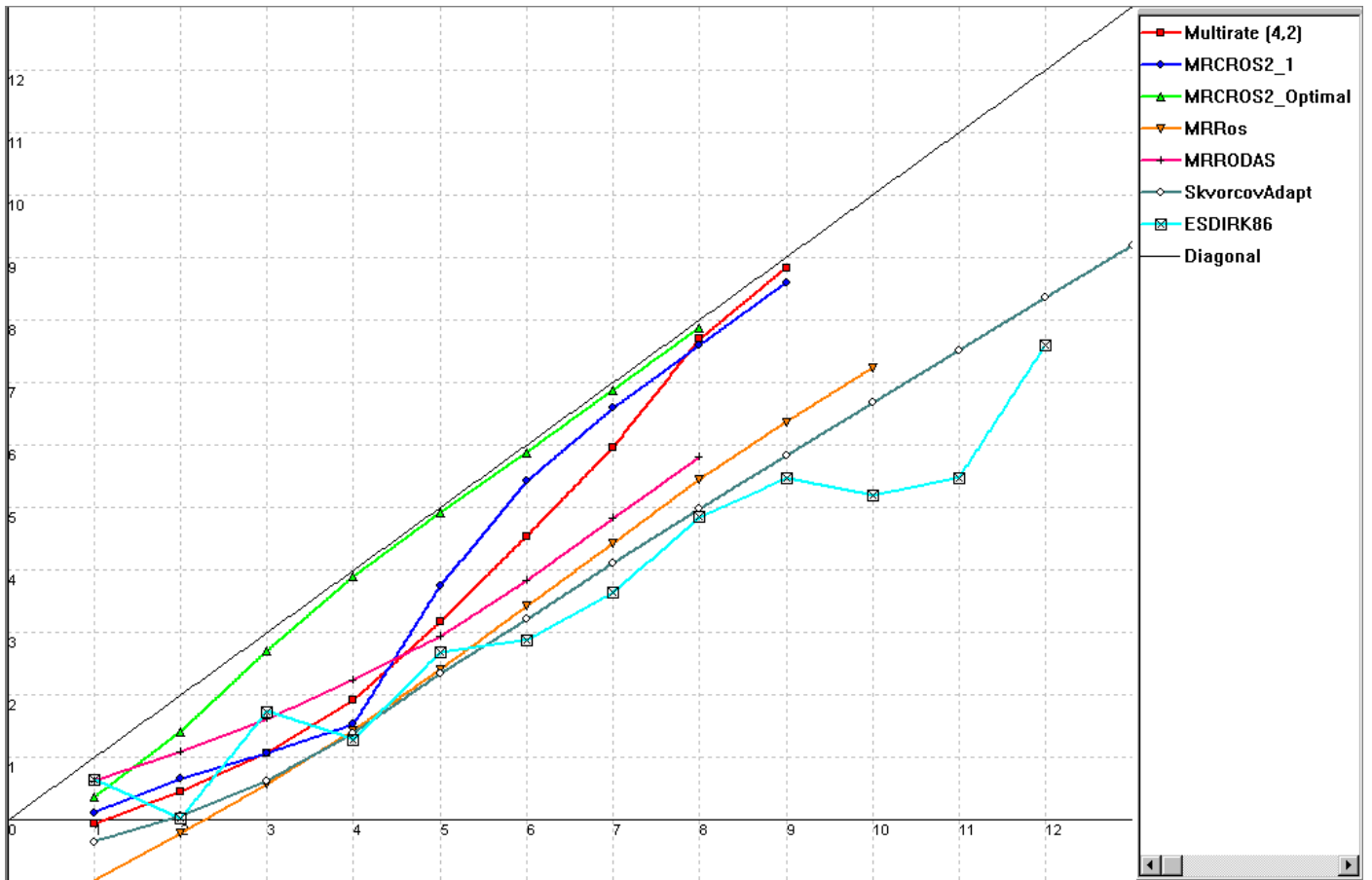


Рисунок 17. Задаваемая точность - достигаемая точность для класса инженерной точности.

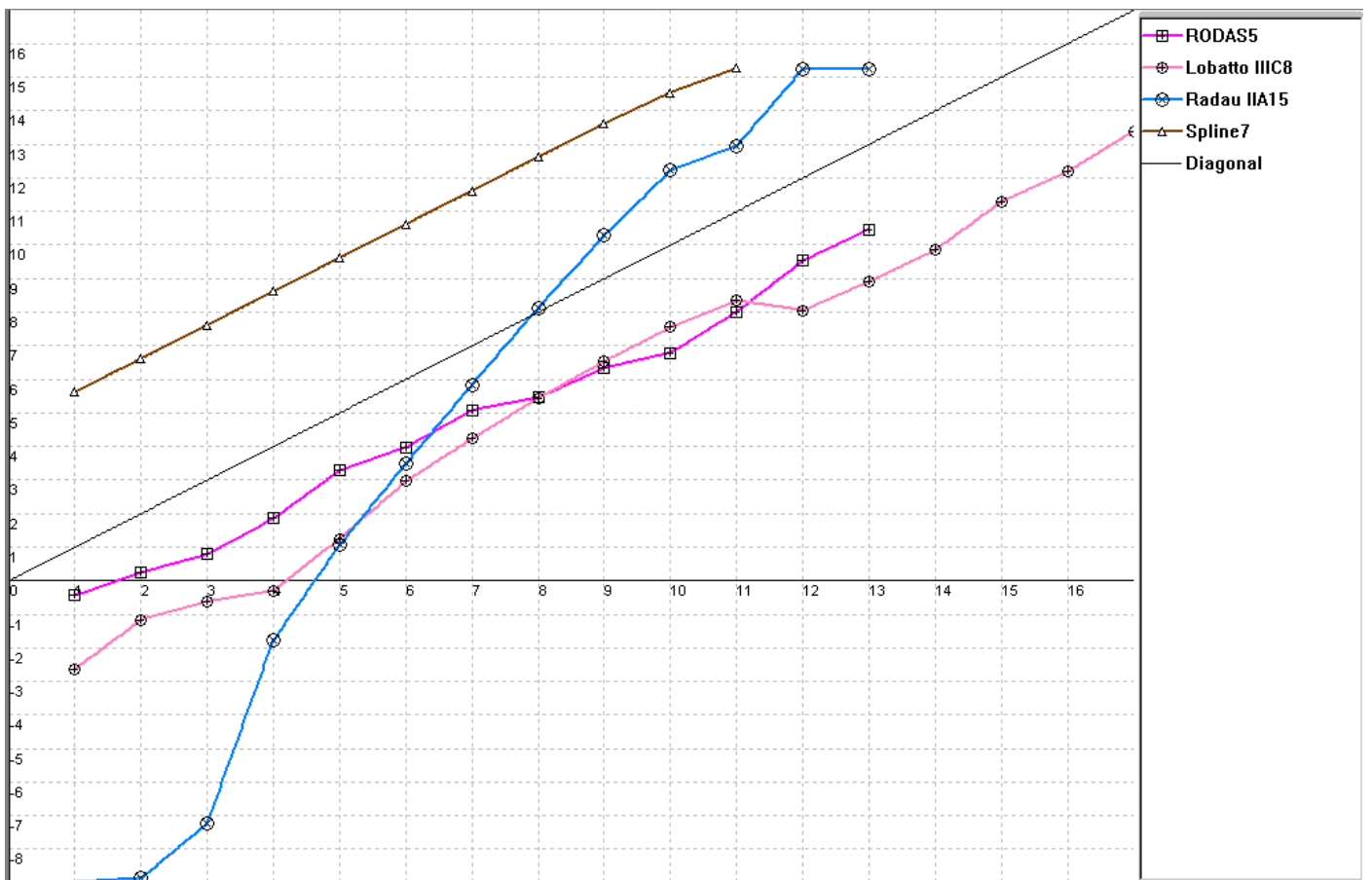


Рисунок 18. Задаваемая точность - достигаемая точность для класса высокой точности.

Список литературы

1. Новиков Е.А. Исследование $(m,2)$ -методов решения жестких систем — Институт вычислительного моделирования СО РАН, Красноярск, Россия: Ж. Вычислительные технологии, том 12, № 5, 2007.//Интернет ресурс <http://cyberleninka.ru/article/n/issledovanie-m-2-metodov-resheniya-zhestkih-sistem.pdf> , 29.12.2017.
2. Пошивайло И.П. Жесткие и плохо обусловленные нелинейные модели и методы их расчета — Диссертация на соискание ученой степени кандидата физико-математических наук: Институт прикладной математики им. М.В. Келдыша — Москва, 2014.//Интернет ресурс http://keldysh.ru/council/3/D00202403/poshivaylo_diss.pdf , 29.12.2017.
3. Лимонов А.Г. Разработка двухстадийных схем Розенброка с комплексными коэффициентами и их применение в задачах моделирования образования периодических наноструктур — Автореферат диссертации на соискание ученой степени кандидата физико-математических наук: Уральский государственный университет им. А.М. Горького — Екатеринбург, 2010.//Интернет ресурс <http://elar.uifu.ru/bitstream/10995/3112/2/urgu0811s.pdf> , 29.12.2017.
4. Ширков П.Д. Оптимально затухающие схемы с комплексными коэффициентами для жестких систем ОДУ — Институт математического моделирования РАН — Москва: Матем. моделирование, 1992, том 4, номер 8, 47–57.//Интернет ресурс <http://mi.mathnet.ru/mm2101> , 29.12.2017.
5. Savcenco V. Construction of a multirate RODAS method for stiff ODEs: Journal of Computational and Applied Mathematics Volume 225, Issue 2, 15 March 2009, Pages 323-337.//Интернет ресурс <http://www.sciencedirect.com/science/article/pii/S0377042708003701> , 29.12.2017.
6. Зубанов А. М., Кутрухин Н. Н., Ширков П. Д. О построении линейно неявных схем, LN-эквивалентных неявным методам Рунге–Кутты: Компьютерные исследования и моделирование 2012 Т. 4 № 3 С. 483-496.//Интернет ресурс http://crm.ics.org.ru/uploads/crmissues/crm_2012_3/483-496.pdf , 29.12.2017.
7. Giovanna A. Di Marzo RODAS5(4) Méthodes de Rosenbrock d'ordre 5(4) adaptées aux problèmes différentiels-algébriques — Mémoire de diplôme en Mathématiques: Université de Genève Faculté des Sciences Section de Mathématiques — Genève, Mars 1993.//Интернет ресурс <http://cui.unige.ch/~dimarzo/papers/DIPL93.pdf> 29.12.2017.
8. Скворцов Л. М. Диагонально-неявные методы Рунге–Кутты для жестких задач: Ж. вычисл. матем. и матем. физ., 2006, том 46, номер 12, С. 2209–2222.//Интернет ресурс <http://mi.mathnet.ru/zvmmf367> , 29.12.2017.
9. Хайрер Э., Ваннер Г. Решение обыкновенных дифференциальных уравнений. Жесткие и дифференциально-алгебраические задачи. — М.: Мир, 1999.
10. Власов С.А., Ширков П.Д. Сплайн-интегрирование дифференциальных уравнений с запаздыванием: Системный анализ в науке и образовании. Университет "Дубна", eISSN: 2071-9612. Дубна 2010.
11. The doubledouble homepage//Интернет ресурс <https://boutell.com/fracster-src/doubledouble/doubledouble.html> , 23.01.2018.
12. David H. Bailey, Xiaoye S. L., Yozo Hida. Quad-Double Arithmetic: Algorithms, Implementation, and Application: October 30, 2000// Интернет ресурс www.davidhbailey.com/dhbpapers/quad-double.pdf , 23.01.2018.
13. Хайрер Э., Нёрсетт С., Ваннер Г. Решение обыкновенных дифференциальных уравнений. Нежесткие задачи. — М.: Мир, 1990.
14. Скворцов Л. М., Явный многошаговый метод численного решения жестких дифференциальных уравнений. — вычисл. матем. и матем. физ., 2007, том 47, номер 6, 959–967//Интернет ресурс <http://mi.mathnet.ru/zvmmf4593> , 22.02.2018.
15. Деккер К., Вервер Я. Устойчивость методов Рунге-Кутты для жестких нелинейных дифференциальных уравнений. — М.: Мир, 1988.
16. Кузнецов А. П., Селиверстова Е. С., Трубецков Д. И., Тюрюкина Л. В. Феномен уравнения ван дер Поля. — Обзор актуальных проблем нелинейной динамики: Изв. вузов «ПНД», т. 22, № 4, 2014. УДК 517.91, 517.938, 51.73 // Интернет ресурс <http://sgtnd.narod.ru/papers/2014PND3.pdf> , 26.03.2018.