

МГТУ им. Н.Э.Баумана

Факультет РК

Кафедра РК-6 (САПР)

к.т.н., доцент Маничев Владимир Борисович:

Тема доклада:

**«ДОСТОВЕРНОЕ МОДЕЛИРОВАНИЕ
ДИНАМИЧЕСКИХ СИСТЕМ С ПОМОЩЬЮ
A($\pi/2$)-УСТОЙЧИВЫХ МЕТОДОВ РЕШЕНИЯ
СИСТЕМ ДИФФЕРЕНЦИАЛЬНО-
АЛГЕБРАИЧЕСКИХ УРАВНЕНИЙ»**

ОБЛАСТЬ ИССЛЕДОВАНИЙ

Численное моделирование (simulation) динамических систем (dynamic systems), математические модели которых описываются системами обыкновенных дифференциальных уравнений (ОДУ) (Ordinary Differential Equations – ODE) или дифференциально-алгебраических уравнений (ДАУ) (Differential-Algebraic Equations - DAE)) **при невысоких по умолчанию требованиям к математической точности решения систем ОДУ-ДАУ (обычно $\epsilon=0.001$)**. Эти невысокие требования объясняются тем, что исходные внутренние параметры динамических систем (коэффициенты систем ОДУ-ДАУ) получены с невысокой математической точностью и имеют технологический разброс.



Ошибки и неточности в компьютерных инженерных вычислениях при численном моделировании проектируемых сложных и опасных технических объектов (авиа-космическая промышленность, ядерная энергетика и др.) могут стать причиной отказов техники, аварий и катастроф

Классическая постановка задач решения систем ОДУ с правой частью

1. Нормальная форма Коши в координатном базисе дифференциальных переменных состояния (явная форма):

$$dX / dt = F(X, t)$$

X - вектор **дифференциальных переменных состояния** размерностью m

F - вектор-функция **правых частей** размерностью m

t - независимая переменная (обычно – время)

Заданы начальные условия $X_0 = X(0)$ и
отрезок интегрирования $t=[0, TK]$.

Классическая постановка задач решения систем ОДУ-ДАУ с правой частью

2. Дифференциально-алгебраическая форма
разных индексов в **полном координатном базисе**
(полуявная форма):

$$\begin{cases} dX / dt = F(X, Y, t) \\ G(X, Y) = 0 \end{cases} \quad \text{или} \quad M \cdot dZ / dt = F(Z, t)$$

Y - вектор **алгебраических переменных**
размерностью k для полного координатного базиса

G - вектор-функция размерностью k

M - сингулярная матрица ранга m размером $(m+k)^2$.

$Z = (X, Y)$ Заданы согласованные начальные условия:

$$X_0 = X(0) \quad Y_0 = Y(0)$$

Классическая постановка задач решения систем ОДУ-ДАУ в полном координатном базисе

3. Дифференциально-алгебраическая форма

в полном координатном базисе (неявная форма)

$$G(X, dX / dt, Y, t) = 0 \quad (1)$$

G - вектор-функция размерностью $m + k$

Заданы согласованные начальные условия и отрезок интегрирования.

Задача (1) была поставлена Линдой Петзольд в 1982 г. и была разработана программа **DASS** – *Differential Algebraic Systems Solver* на основе замены вектора производных в (1) по формулам дифференцирования назад (**ФДН**, *Backward Differential Formula (BDF)*) и решении соответствующих систем нелинейных алгебраических уравнений (НАУ):

$$dX / dt = BDF(X, h) \quad h - \text{шаг интегрирования}$$

Постановка задачи решения систем ОДУ-ДАУ в расширенном координатном базисе переменных

$$G(X, PX, Y, t) = 0 \quad (2)$$

X - вектор дифференциальных переменных состояния размерностью m

$PX = dX / dt$ - вектор производных дифференциальных переменных по времени размерностью m

Y - вектор алгебраических переменных размерностью k

t - независимая переменная (обычно время)

G - вектор-функция размерностью $m + k$

Заданы начальные условия и отрезок интегрирования.

Найти $X(t), Y(t), PX(t)$ с гарантированной достоверностью и точностью для всех известных тестовых задач.

Алгоритм решения систем ОДУ-ДАУ в расширенном координатном базисе переменных на примере неявного метода Эйлера

На каждом шаге интегрирования h_n для момента времени t_n формируется система НАУ H и G числом уравнений $2m + k$

$$\begin{cases} H(PX_n, X_n, X_{n-1}, h_n) = PX_n - (X_n - X_{n-1}) / h_n = 0 \\ G(X_n, PX_n, Y_n, t_n) = 0 \end{cases}$$

относительно векторов X_n, PX_n, Y_n общей размерности $2m+k$

H - система алгебраических уравнений численного интегрирования с числом уравнений m , соответствующая формуле неявного метода Эйлера

Система НАУ (H, G) решается методом Ньютона-Рафсона, который сводится к решению соответствующих систем линейных алгебраических уравнений (ЛАУ) на каждом шаге интегрирования h_n

$$h_n = t_n - t_{n-1}, \quad n - \text{номер шага интегрирования}$$

**Системы алгебраических уравнений H_i для
новых S -стадийных DAbs методов интегрирования
высокого порядка точности**

(развитие Ab^Tc известных методов Бутчера Д.)

$$H_i(PX_i, X_i, X_{n-1}, X_{n-1}, h_n) =$$

$$= h_n \sum_{j=1}^s d_{ij} PX_j - \sum_{j=1}^s a_{ij} X_j - b_i X_{n-1} - h_n c_i PX_{n-1} = 0$$

$X_n = X_s, PX_n = PX_s$, номер стадии $i = 1, \dots, s, t_n = t_s$,

s – число стадий, d_{ij}, a_{ij}, b_i, c_i – параметры метода

Правая часть системы ОДУ не нужна!

Если матрица **D** единичная, то новые методы будут соответствовать методам Рунге-Кутты из работ Бутчера Д.

Схема алгоритма 1-но стадийных неявных методов 1-го и 2-го порядков точности (M_1 и M_2 число стадий $S=1$)

$$\begin{array}{ccc} X_{n-1}, & \text{система НАУ:} & X_n = X_1, \\ Y_{n-1}, & \left\{ \begin{array}{l} H_1 = 0 \\ G_1 = 0 \end{array} \right. & Y_n = Y_1, \\ PX_{n-1} & \longrightarrow & PX_n = PX_1 \end{array}$$

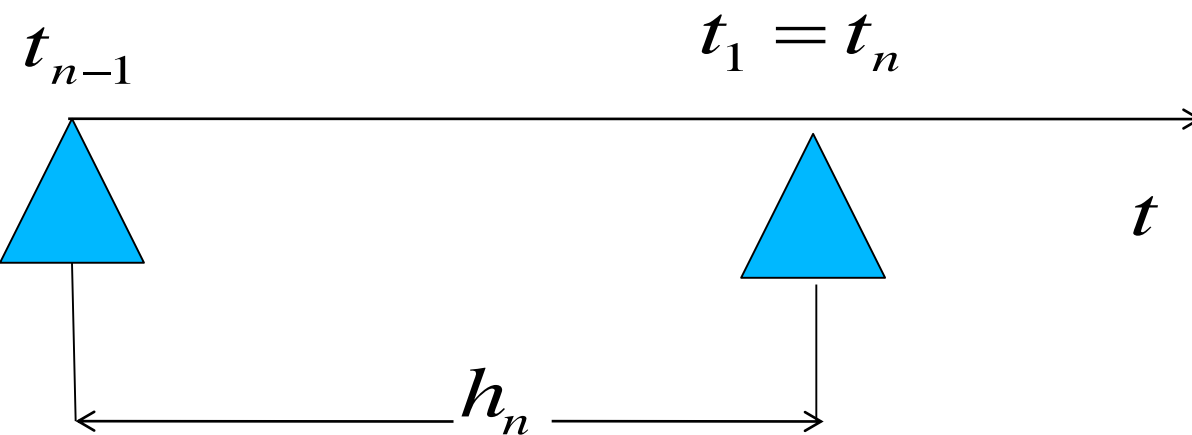
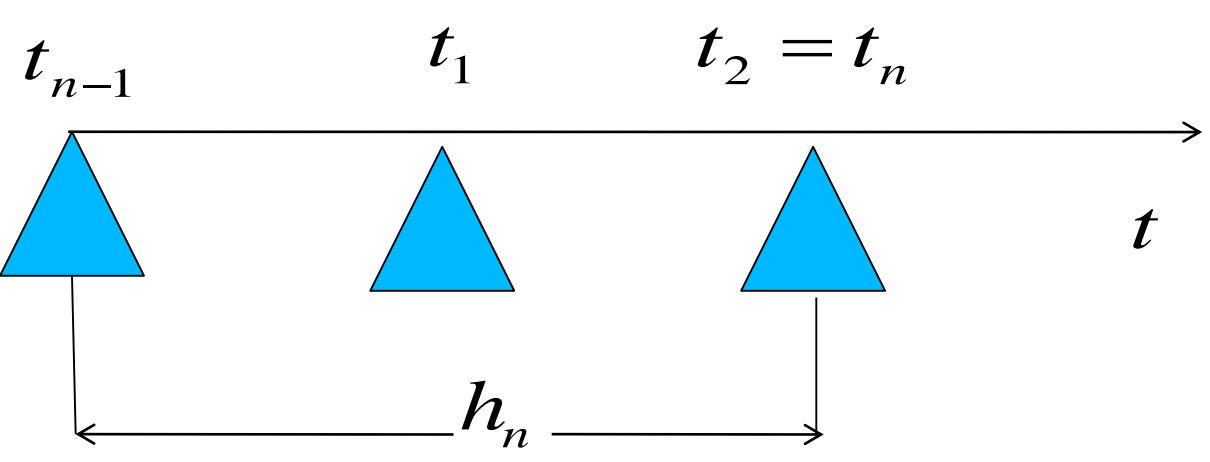


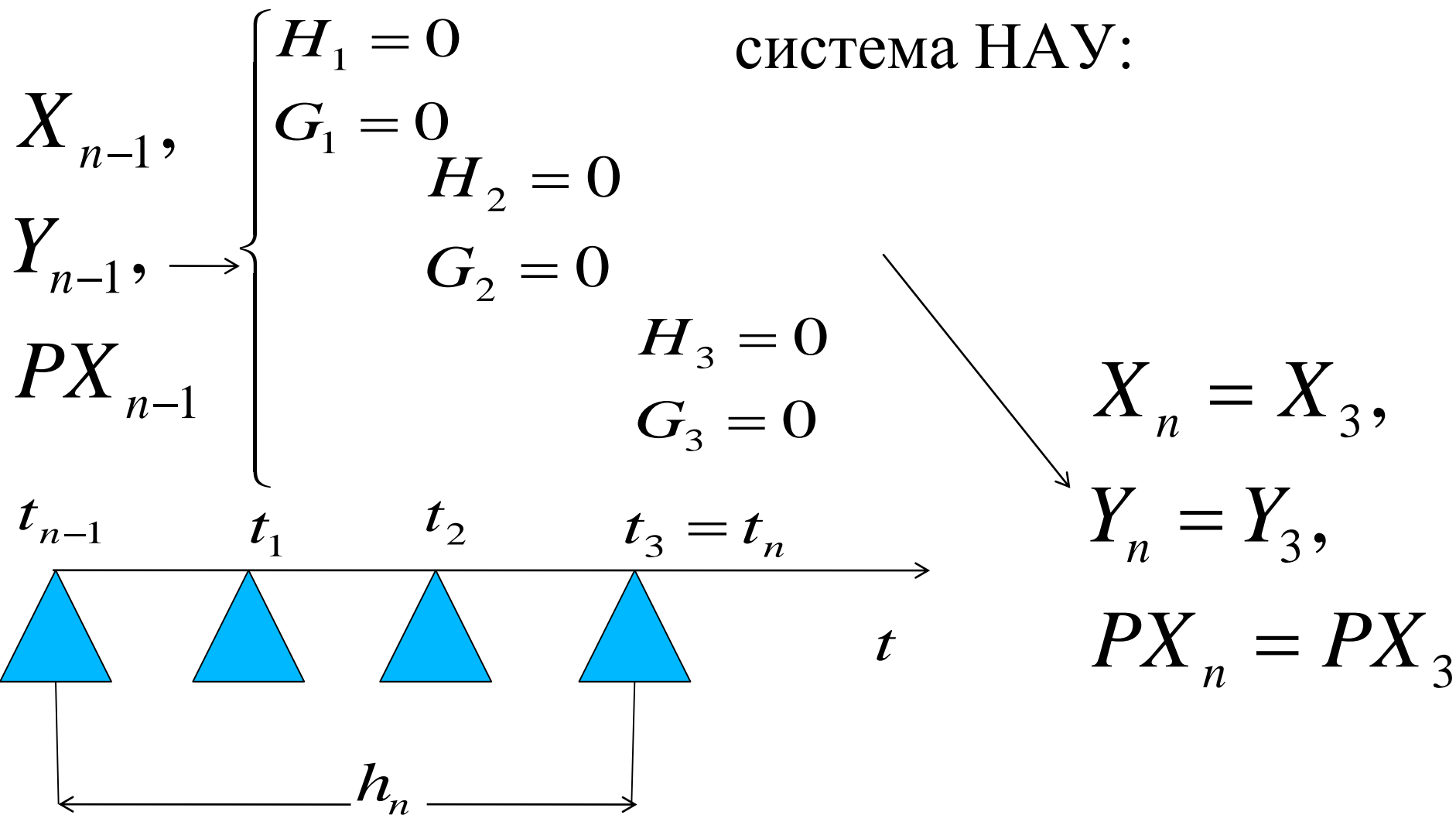
Схема алгоритма неявного 2-х стадийного метода 4-го порядка точности (M_4 число стадий $S=2$)

$$\begin{matrix} X_{n-1}, \\ Y_{n-1}, \\ PX_{n-1} \end{matrix} \longrightarrow \left\{ \begin{array}{l} H_1 = 0 \\ G_1 = 0 \\ H_2 = 0 \\ G_2 = 0 \end{array} \right. \quad \text{система НАУ:}$$



$$\begin{aligned} X_n &= X_2, \\ Y_n &= Y_2, \\ PX_n &= PX_2 \end{aligned}$$

Схема алгоритма неявного 3-х стадийного метода 6-го порядка точности (M_6 число стадий $S=3$)



Основной недостаток известных программно-методических комплексов (ПМК) моделирования динамических систем (на примере MATLAB-SIMULINK)

Получение неверного результата численного моделирования динамических систем при невысоких, заданных по умолчанию, требованиях к **eps** -математической точности решения систем ОДУ-ДАУ, (по умолчанию **eps=0.001** в программах решения систем ОДУ-ДАУ в MATLAB-SIMULINK)
(слайды ниже)

Выбор системы H на основе 5-ти базовых классов задач решения уравнения Далквиста

Уравнение Далквиста (1963 г.)

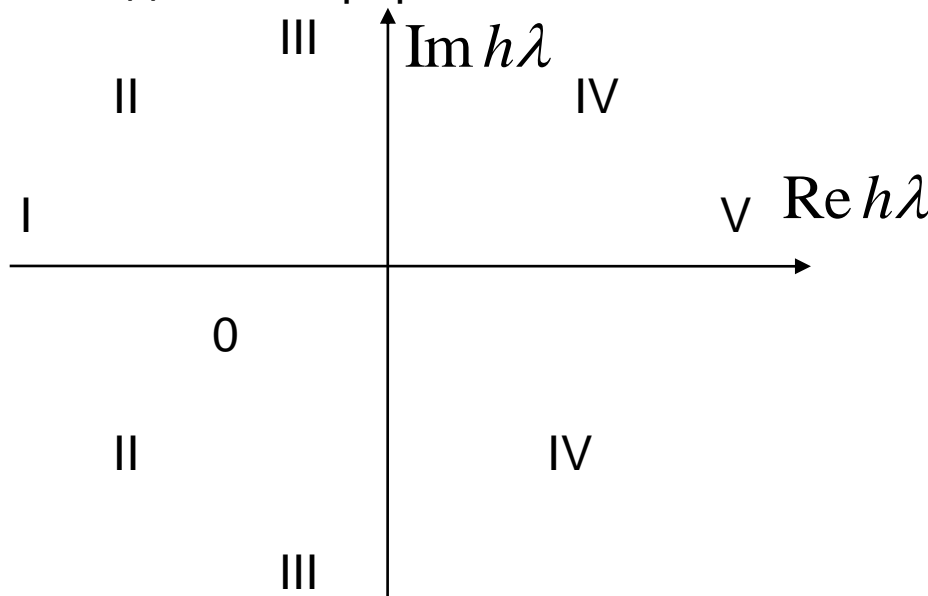
для исследования устойчивости методов интегрирования

$$dy / dt = \lambda y \quad \lambda = \text{Re } \lambda + j \text{Im } \lambda \quad - \text{ комплексный коэффициент}$$

$$y_0 = 1 \quad - \text{ начальные условия} \quad y(t) = e^{\lambda t} \quad - \text{ аналитическое решение}$$

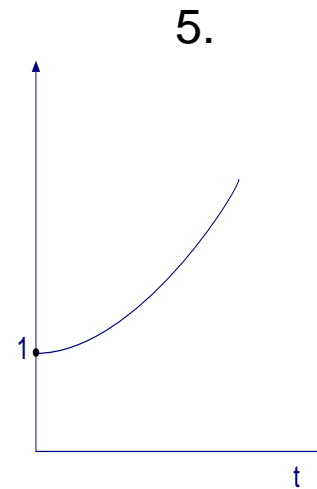
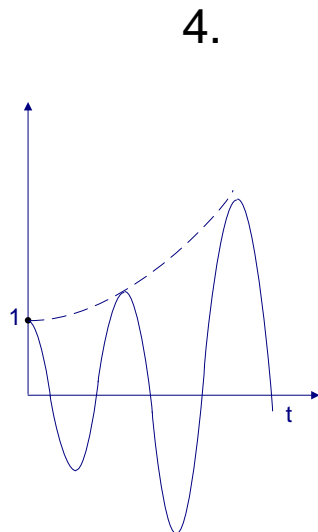
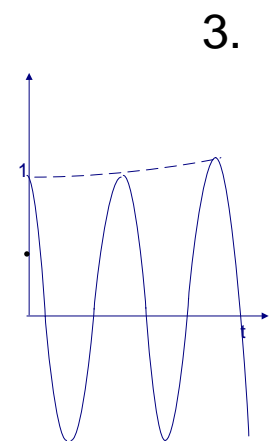
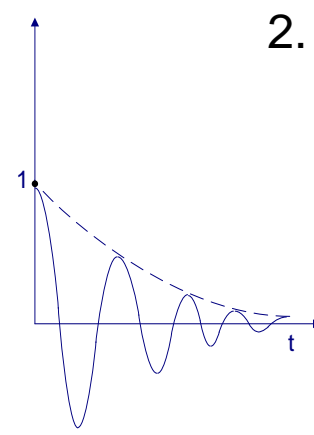
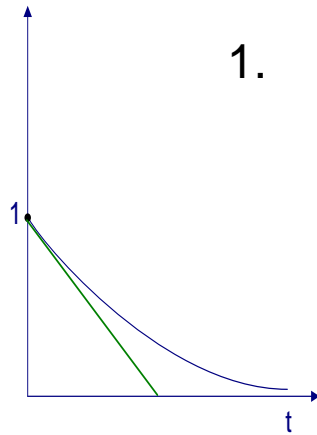
Комплексная плоскость устойчивости
методов интегрирования

h - постоянный шаг интегрирования



1. Фундаментальное решение лежит в области I
2. Фундаментальное решение лежит в области II
3. Фундаментальное решение лежит в области III
4. Фундаментальное решение лежит в области IV
5. Фундаментальное решение лежит в области V

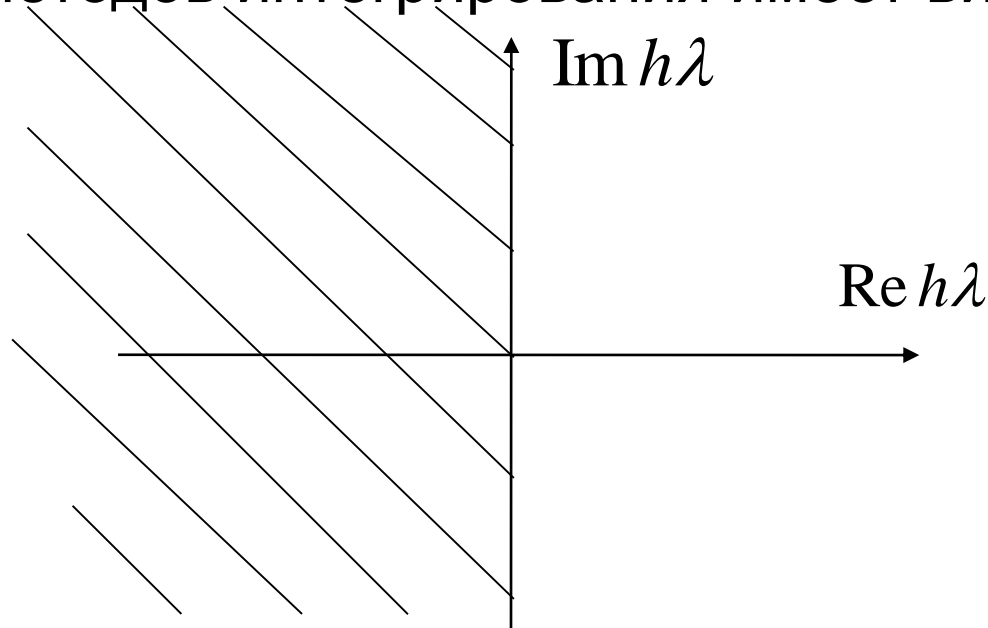
Фундаментальные решения линейных систем ОДУ для классов задач 1,2,3,4,5



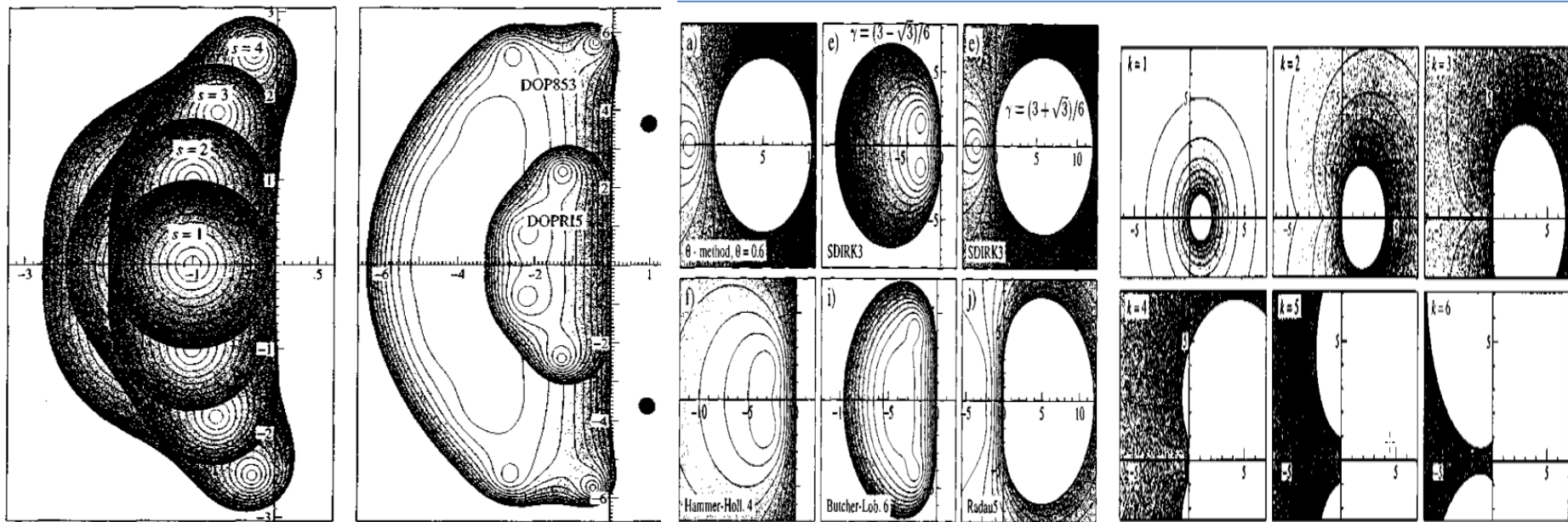
Обеспечение **ДОСТОВЕРНОСТИ** - основное требование к методам численного интегрирования систем ОДУ-ДАУ при **невысоких требованиях к точности интегрирования**

При невысокой заданной точности интегрирования шаг интегрирования может стать таким большим, что не $A(\pi/2)$ -устойчивые методы интегрирования могут терять устойчивость.

Область абсолютной устойчивости $A(\pi/2)$ -устойчивых методов интегрирования имеет вид:



Области абсолютной устойчивости классических явных и неявных методов решения систем ОДУ-ДАУ (реализован только классический $A(\pi/2)$ -устойчивый неявный метод трапеций 2-го порядка точности)



а)

б)

а) – явные методы Рунге-Кутты и Дормана-Принса б) – классические неявные методы Рунге-Кутты и Гира-Петзольд (ФДН)

При малых шагах интегрирования (окрестность точки 0,0) все методы будут $A(\pi/2)$ -устойчивыми и решение будет достоверным, но при невысоких требованиях к точности интегрирования шаг интегрирования может стать таким большим, что классические методы интегрирования могут потерять устойчивость (область устойчивости заштрихована)

Кардинальные проблемы реализации $A(\pi/2)$ -устойчивых S-стадийных методов интегрирования

1. **Главный недостаток $A(\pi/2)$ -устойчивых методов – наличие «ложных (паразитных) колебаний» (“ringing”), поэтому Хайрер в книгах (1996, 1999) для жестких систем ОДУ-ДАУ назвал такую устойчивость **нежелательной** (1999, рис. 3.2. стр. 58 – **следующий слайд**), поэтому эти методы не нашли широкого применения на практике. Мы определили причину «ложных» колебаний: разрывы производных в интегрируемых функциях и значительное увеличение шага интегрирования и решили эту проблему.**
2. **Проблема решения систем ЛАУ с повышенной точностью вычислений (extra precision calculations).** Эта проблема была нами поставлена и решена в **2010-2016** годах.

Последний предел должен тогда равняться 1 по модулю, так как $|R(iy)|=1$ для всех вещественных y . Это означает, что для z , близких к вещественной оси и с очень большой отрицательной вещественной частью, $|R(z)|$, хотя и <1 , но очень близко к единице. Как следствие, жесткие компоненты в (2.6) демпфируются, но очень медленно. Покажем это на примере задачи

$$y' = -2000(y - \cos x), \quad y(0) = 0, \quad 0 \leq x \leq 1.5 \quad (3.11)$$

— такой же, что и (1.1), но с большей жесткостью. На рис. 3.2 численные результаты для правила трапеций даны в сравнении с результатами для неявного метода Эйлера. Неявный метод Эйлера демпфирует переходную фазу много быстрее, чем правило трапеций. Таким образом, представляется желательным свойство метода, состоящее в том, чтобы $|R(z)|$ была много меньше чем 1 при $z \rightarrow -\infty$.

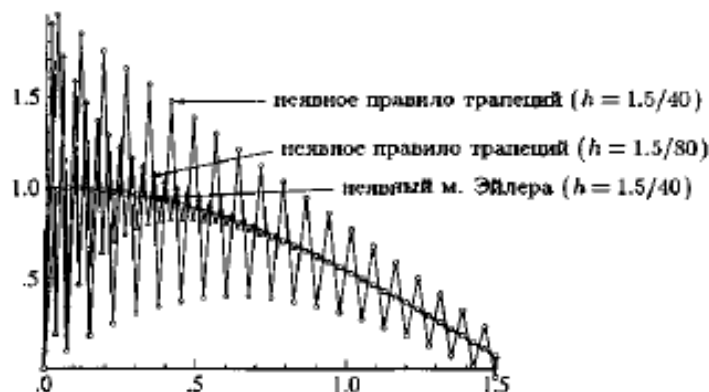


Рис. 3.2. Правило трапеций в сравнении с неявным методом Эйлера, решение задачи (3.11).

Определение 3.7 (Ил 1969). Метод называется L -устойчивым, если он A -устойчивый и если вдобавок

$$\lim_{z \rightarrow -\infty} R(z) = 0. \quad (3.12)$$

Среди методов из табл. 3.1 L -устойчивыми являются неявный метод Эйлера, ОДНРК-метод с $\gamma = (2 \pm \sqrt{2})/2$ и формула Радо ПА (j).

Предложение 3.8. Если неявный метод Рунге—Кутты с невырожденной матрицей A удовлетворяет одному из следующих условий:

$$a_{s,j} = b_j, \quad j = 1, \dots, s, \quad (3.13)$$

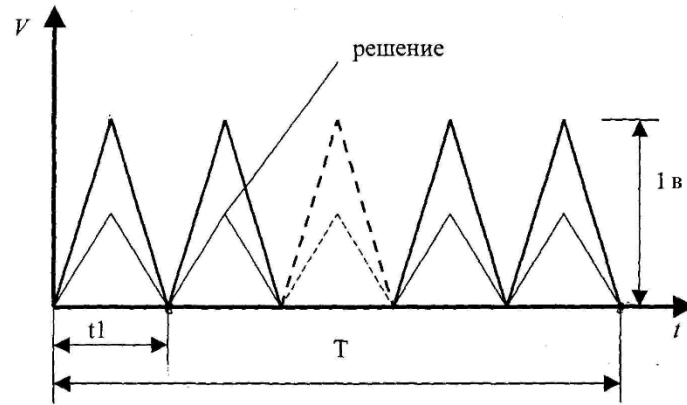
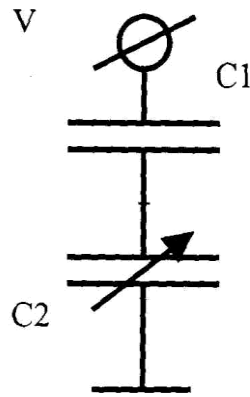
$$a_{i1} = b_1, \quad i = 1, \dots, s, \quad (3.14)$$

то $R(\infty)=0$. Это делает A -устойчивые методы L -устойчивыми.

Проблема 1. Алгоритм реализации $A(\pi/2)$ - устойчивых методов интегрирования без «ложных» колебаний

Если в начале или в пределах текущего шага интегрирования $A(\pi/2)$ - устойчивого метода интегрирования присутствуют моменты времени, соответствующие разрывам производных интегрируемых функций, то в конце текущего шага выполняется **коррекция значений векторов X , PX и Y** путем выполнения корректирующего шага интегрирования, равному текущему шагу интегрирования по неявному методу Эйлера.

Проверка алгоритма на тестовой задаче «емкостной делитель напряжения»



Математическая модель данной тестовой задачи в форме исходной (полученной по законам физики) системы ДАУ имеет вид:

$$\begin{cases} C1(dU_{C1}(t)/dt) - i(t) = 0 \\ C1(0.5 - U_{C2}(t)) \times (dU_{C2}(t)/dt) - i(t) = 0 \\ U_{C1}(t) + U_{C2}(t) - V(t) = 0 \end{cases}$$

$$i(t) = i_{C1}(t) = i_{C2}(t)$$

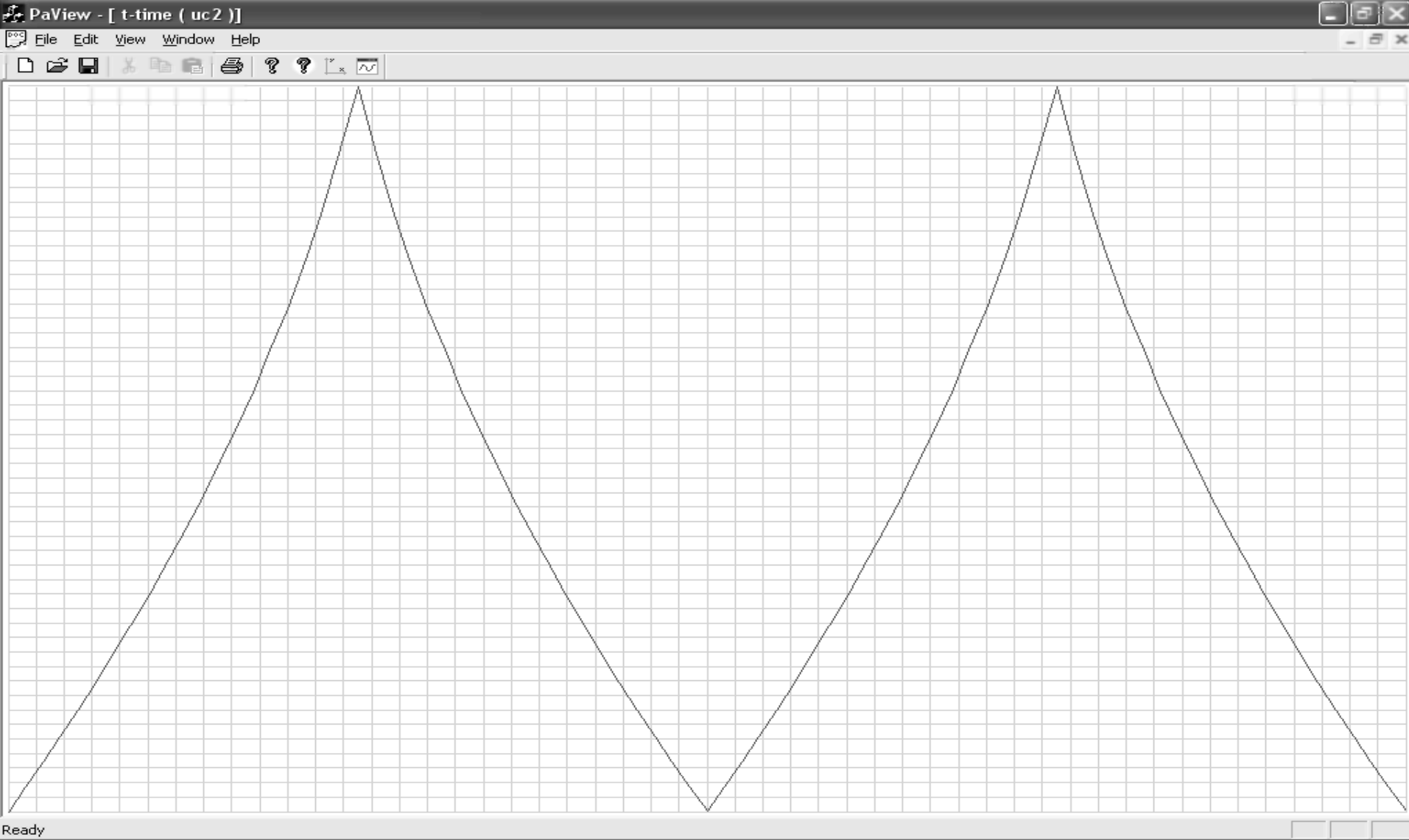
Аналитическое решение этой задачи для напряжения и тока на емкости C2 при начальных условиях :

$$U_{C1}(0) = 0 \quad U_{C2}(0) = 0$$

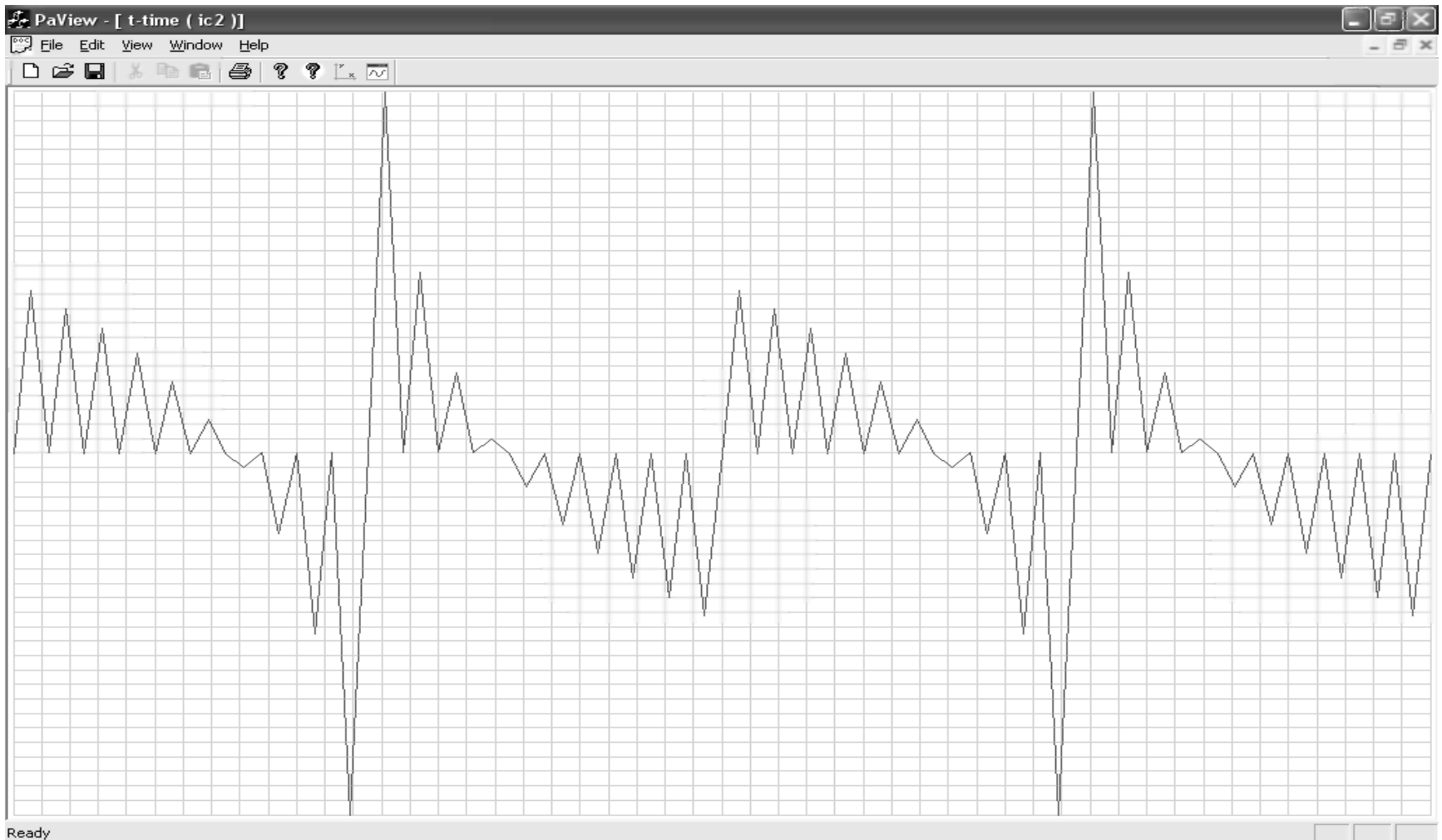
$$U_{C2}(t) = C1 + C2(0) - \sqrt{((C1 + C2(0))^2 - 2C1V(t))}$$

$$i_{C2}(t) = \frac{C1(0.5 - U_{C2}(t))}{\sqrt{(C1 + C2(0))^2 - 2C1V(t)}}$$

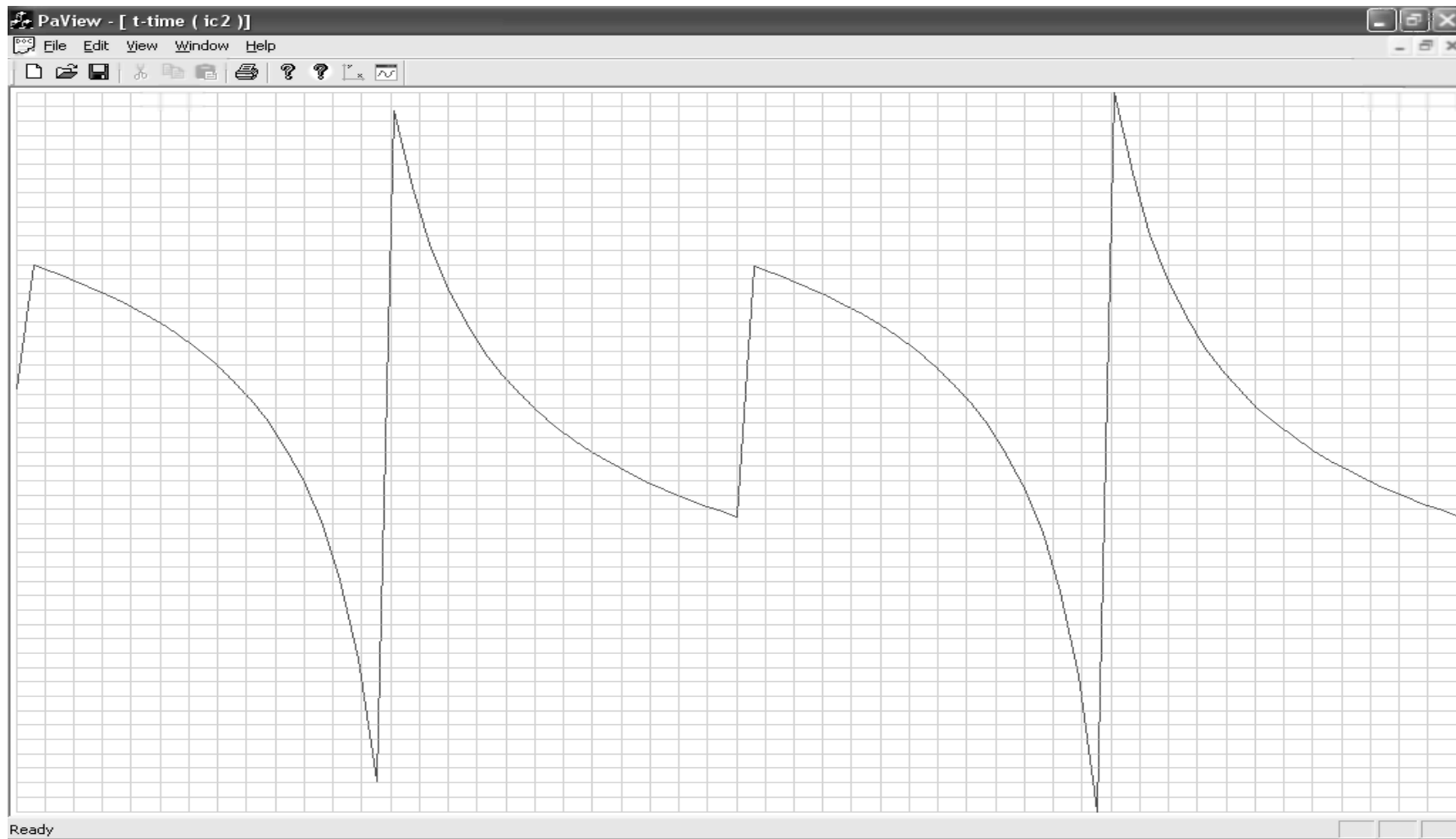
Корректное и точное решение для напряжения $u_{c2}(t)$, рассчитанного классическим $A(\pi/2)$ - устойчивым неявным методом трапеций



«Ложные (паразитные) колебания» для тока $i_{c2}(t)$, рассчитанного классическим неявным методом трапеций



Ток $i_{c2}(t)$, рассчитанный $A(\pi/2)$ - устойчивым методом по рассмотренному алгоритму



Проблема 2. Постановка задачи решения систем ЛАУ с повышенной точностью вычислений (extra precision calculations)

$$Ax = b$$

$A = (a_{ij})$ – матрица коэффициентов размером $n \times n$,
 b – вектор-столбец правых частей, размерностью n ,
 x - вектор неизвестных размерностью n .

Необходимо вычислить **все элементы вектора x** с гарантированной точностью в **15 верных значащих цифр** (общепринятая двойная точность арифметических операций на компьютерах: точность **double precision** на алгоритмических языках ФОРТРАН и Си, стандарт IEEE-754-2008)

Сравнение программ-решателей систем ЛАУ

ПРОГРАММА-РЕШАТЕЛЬ СИСТЕМ ЛАУ	ТЕСТОВАЯ ЗАДАЧА		
	ТЕСТОВАЯ ЗАДАЧА УИЛКИНСОНА (WILKINSON) 2-ГО ПОРЯДКА ОБУСЛОВЛЕННОСТЬ СИСТЕМЫ $COND(A)=10^6$	ТЕСТОВАЯ ЗАДАЧА ВОХМИНЦЕВА 5-ГО ПОРЯДКА ОБУСЛОВЛЕННОСТЬ СИСТЕМЫ $COND(A)=10^{12}$	ТЕСТОВАЯ ЗАДАЧА С МАТРИЦЕЙ ГИЛЬБЕРТА (HILBERT) 10-ГО ПОРЯДКА ОБУСЛОВЛЕННОСТЬ СИСТЕМЫ $COND(A)=1.5*10^{13}$
<i>SADEL</i>	+ (15)	+ (15)	+ (15)
<i>MATLAB</i>	- (8)	- (5)	- (4)
<i>MATHCAD</i>	- (8)	- (5)	- (4)
<i>MATHEMATICA</i>	- (8)	- +(5)	-+ (4)
<i>MAPLE</i>	- (8)	- (5)	- (н)
<i>NAG-LAPACK</i>	- (8)	- (5)	- (н)
<i>IMSL</i>	- (8)	- (5)	- (н)
<i>INTEL MKL</i>	- (8)	- (5)	- (н)
<i>MAGMA-LAPACK</i>	- (8)	- (5)	- (4)

Знак (–) означает, что решатель систем ЛАУ не обеспечил точность в **15 верных значащих цифр** для всех элементов вектора решений теста и не выдал (–+ выдано предупреждение) никакого предупреждающего сообщения об этом, в скобках (*m*) указано полученное количество верных значащих цифр в решении теста. **н** – нет данных

**Пример. Тестовая задача с матрицей Гильберта
(*Hilbert matrix*) 10-го порядка (все элементы
матрицы **A** и вектора **B**, как суммы
соответствующих строк матрицы **A**, вычислены с
четверной точностью)**

Вектор B – построчная сумма коэффициентов:

Матрица A

$$A_{ij} = 1/(i + j - 1)$$

$$i, j = 1, \dots, 10$$

i=1 Bi=	2.928968253968250
i=2 Bi=	2.019877344877340
i=3 Bi=	1.603210678210670
i=4 Bi=	1.346800421800420
i=5 Bi=	1.168228993228990
i=6 Bi=	1.034895659895660
i=7 Bi=	0.930728993228990
i=8 Bi=	0.846695379783620
i=9 Bi=	0.777250935339180
i=10 Bi=	0.718771403175440

**Абсолютно точное решение этой тестовой
задачи: Единичный вектор**

Результаты решения с удвоенной точностью на языке Си тестовой задачи с матрицей Гильберта (Hilbert matrix) 10-го порядка

Си библиотека CUBLAS-MAGMA
(LAPACK-Linpack):

1.0000000000**464697**
0.9999999**60244827**
1.000000**840984605**
0.99999**2393461859**
1.0000**36137385782**
0.9999**00979283788**
1.000**162025520190**
0.999**843779478675**
1.0000**81852296234**
0.9999**82030574678**

Си библиотека SADEL:

i=1 Xi= 1.0000000000000000
i=2 Xi= 1.0000000000000000
i=3 Xi= 1.0000000000000000
i=4 Xi= 1.0000000000000000
i=5 Xi= 1.0000000000000000
i=6 Xi= 1.0000000000000000
i=7 Xi= 1.0000000000000000
i=8 Xi= 1.0000000000000000
i=9 Xi= 1.0000000000000000
i=10 Xi= 1.0000000000000000

***КУРСИВОМ ВЫДЕЛЕНА НЕВЕРНЫЕ ЗНАЧАЩИЕ ЦИФРЫ В РЕШЕНИИ
ТЕСТОВОЙ ЗАДАЧИ С МАТРИЦЕЙ ГИЛЬБЕРТА***

Сравнение программ-решателей систем ОДУ-ДАУ с решателем *manzhuk* из библиотеки SADEL (предыдущая библиотека перед MZK) при невысоких требованиях к точности интегрирования ($\epsilon=0.001$)

Программа- решатель систем ОДУ-ДАУ	ТЕСТЫ					
	ТЕСТ 1	ТЕСТ 1	ТЕСТ 2	ТЕСТ 2	ТЕСТ 3	ТЕСТ 4
	Жесткость MU=1e6	Жесткость MU=1e9	Коэффициенты kt=1, ki=1, ku=0.01	Коэффициенты ku=1, ki=1, kt=1e-104	Жесткость MU=1e6	Параметры реального лазера
SADEL C-Library Метод МЗ	+	+	+	+	+	+
<i>NAG C-Library Метод BDF</i>	+	-	-	-	-	+
<i>IMSL C-Library Метод BDF</i>	+	-	-	-	-	-

Знак (-) означает недостоверное решение теста без предупреждения

Си библиотека *Intel MKL* не имеет программ-решателей систем ОДУ-ДАУ

Сравнение программ-решателей систем ОДУ-ДАУ (продолжение)

Программа- решатель систем ОДУ-ДАУ	ТЕСТЫ					
	ТЕСТ 1	ТЕСТ 1	ТЕСТ 2	ТЕСТ 2	ТЕСТ 3	ТЕСТ 4
	Жесткость MU=1e6	Жесткость MU=1e9	Коэффициенты kt=1, ki=1, ku=0.01	Коэффициенты ku=1, ki=1, kt=1e-104	Жесткость MU=1e6	Параметры реального лазера
SADEL C-Library Метод МЗ	+	+	+	+	+	+
Mathcad метод Radau	+	-	-	-	+	+
MATLAB Метод ODE15s	+	-	-	-	-	+
Maple Метод Rosenbrock	-	-	+	-	-	-
Mathematica Метод BDF	-	-	+	+	+	+

Знак (-) означает недостоверное решение теста без предупреждения

ТЕСТ 1 Пример расчета жесткой системы ОДУ 2-го порядка (тест Ван дер Поля)

$$dx_1 / dt = x_2$$

$$dx_2 / dt = -x_1 + MU \times (1 - x_1^2) \times x_2$$

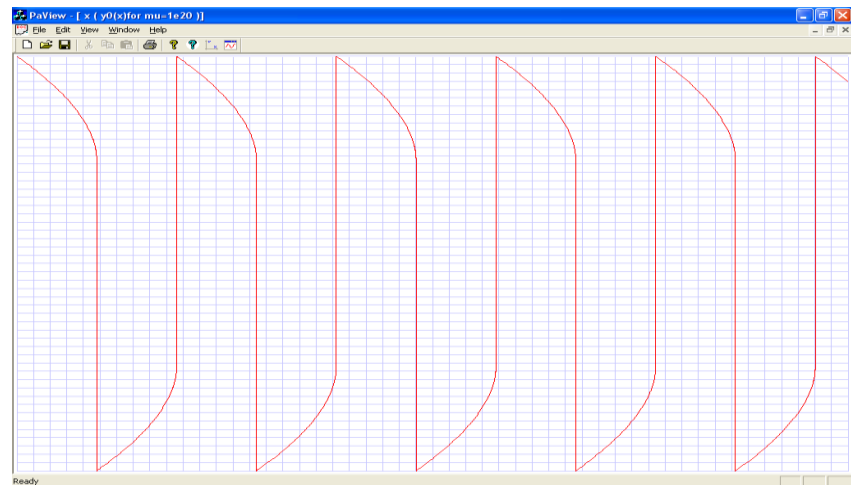
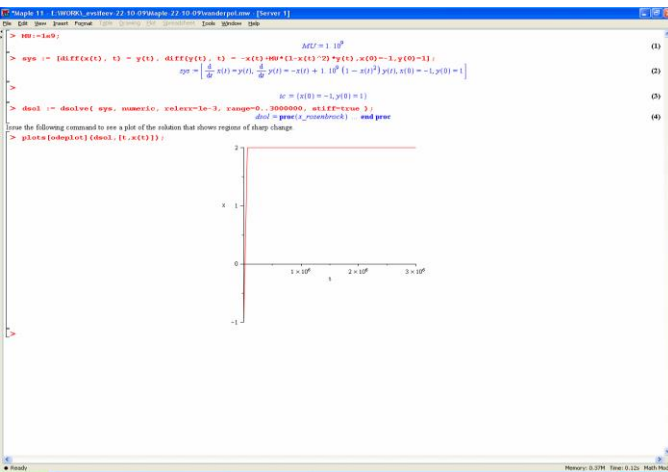
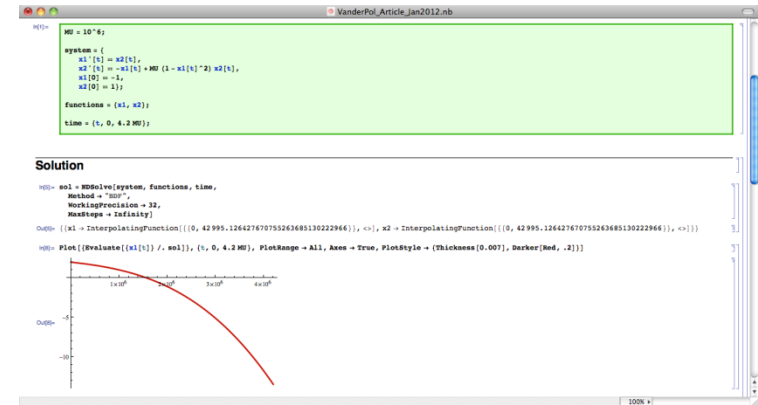
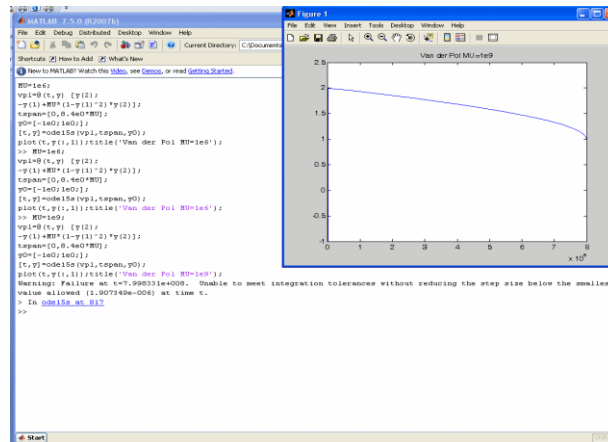
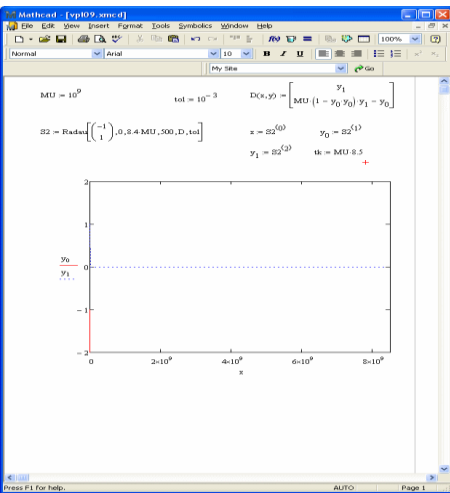
$$x_1(0) = -1, x_2(0) = 1,$$

$$t \in (0, 4.2 \times MU)$$

MU – параметр жесткости.

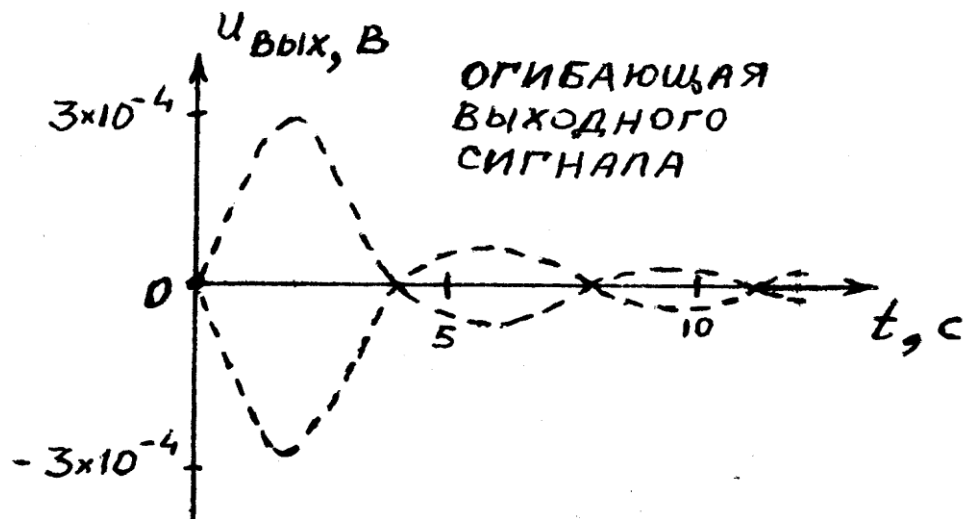
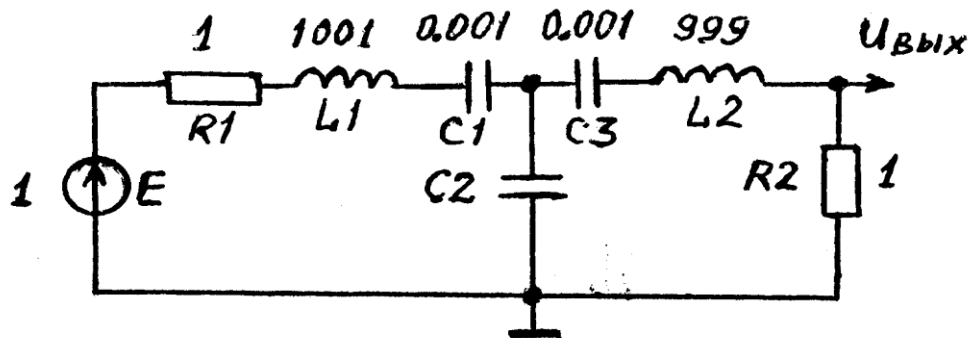
Сравнение было проведено для часто встречающихся на практике параметров жесткости $MU=10^6$ и $MU=10^9$

ОПИСАНИЯ, НЕКОРРЕКТНЫЕ РЕШЕНИЯ ТЕСТА 1 В MATHCAD, MATLAB, MATHEMATICA, MAPLE И КОРРЕКТНОЕ РЕШЕНИЕ В SADEL



ТЕСТ 2. ЛИНЕЙНАЯ ЭЛЕКТРИЧЕСКАЯ СХЕМА С ИЗВЕСТНЫМ АНАЛИТИЧЕСКИМ МНОГО-ПЕРИОДНЫМ РЕШЕНИЕМ (высокооборотный фильтр)

RLC: high Q filter



Система ОДУ для ТЕСТА 2

$$kr = ku / ki, kc = kt \cdot ki / ku, kl = kt \cdot ku / ki$$

$$dx_1 / dt = x_4 / 0.001 \cdot kc$$

$$dx_2 / dt = x_5 / 0.001 \cdot kc$$

$$dx_3 / dt = (x_4 - x_5) / kc$$

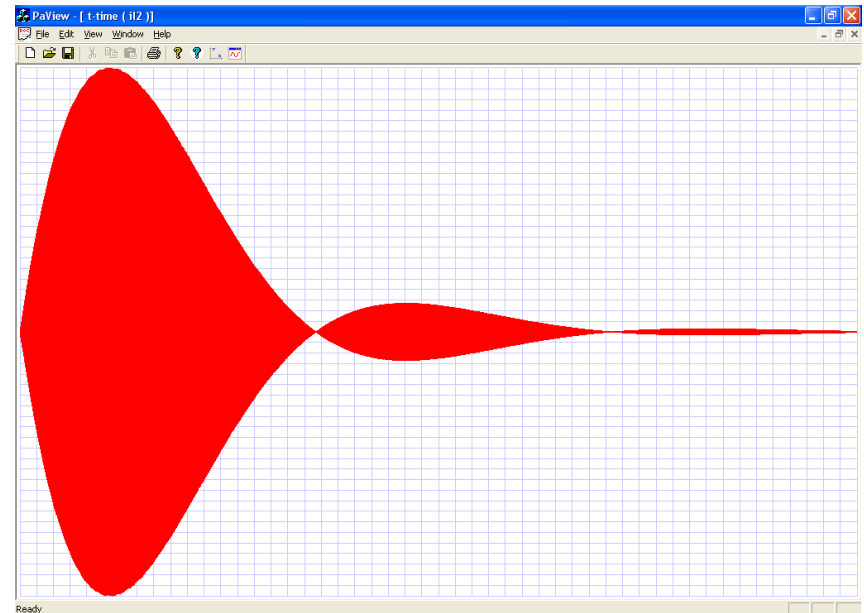
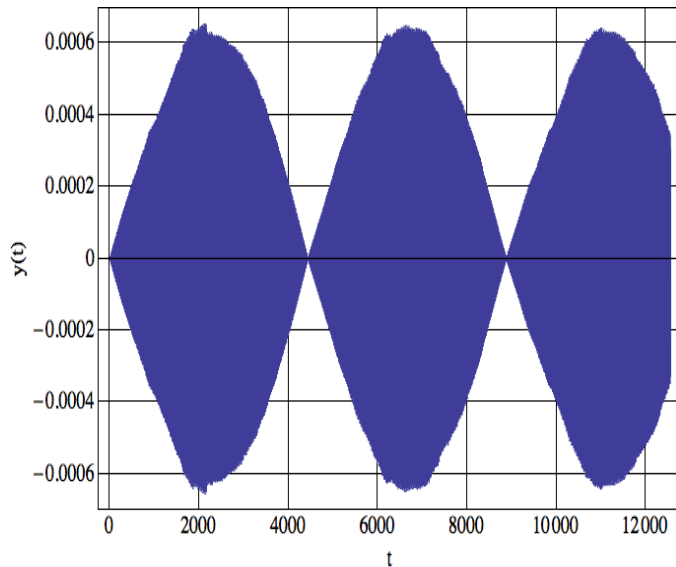
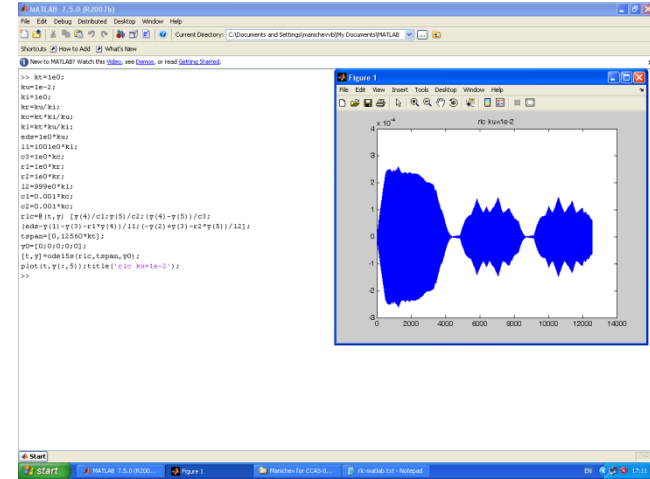
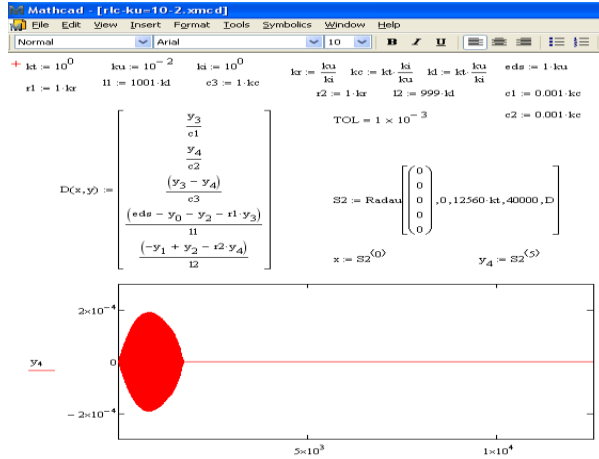
$$dx_4 / dt = (ku - x_1 - x_3 - kr \cdot x_4) \cdot / 1001 \cdot kl$$

$$dx_5 / dt = (-x_2 + x_3 - kr \cdot x_5) / 999 \cdot kl$$

$$x_1(0) = 0, x_2(0) = 0, x_3(0) = 0, x_4(0) = 0, x_5(0) = 0$$

$$t \in (0, 12560 \cdot kt)$$

ОПИСАНИЯ, НЕКОРРЕКТНЫЕ РЕШЕНИЯ ТЕСТА 2 в MATHCAD, MATLAB, C-Library NAG и КОРРЕКТНОЕ РЕШЕНИЕ в Си библиотеке SADEL



ТЕСТ 3 Нелинейная жесткая система ОДУ, имеющая локально-неустойчивое решение

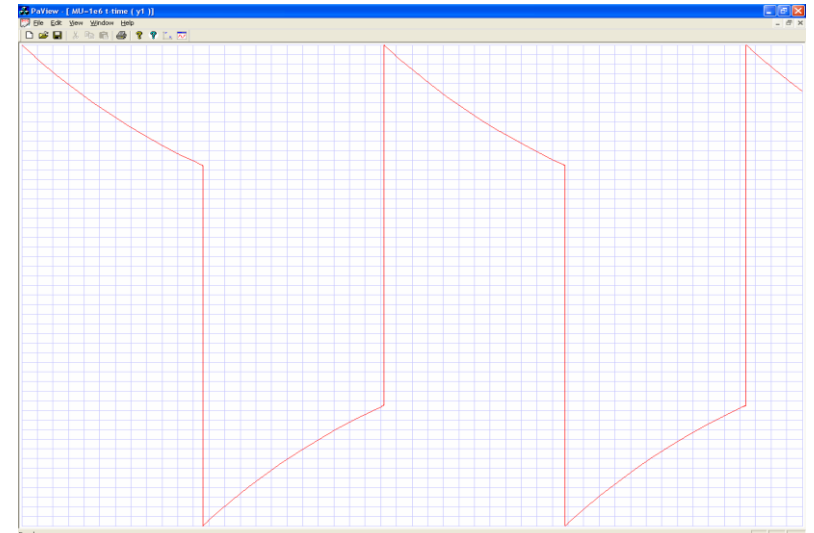
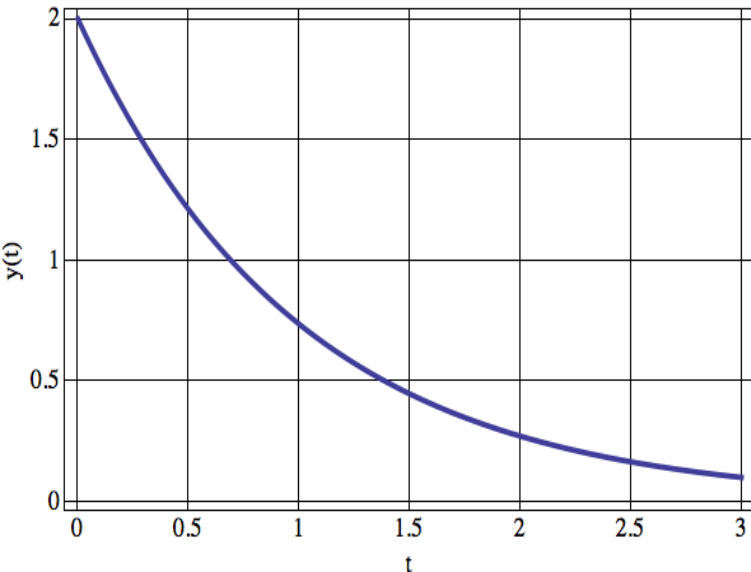
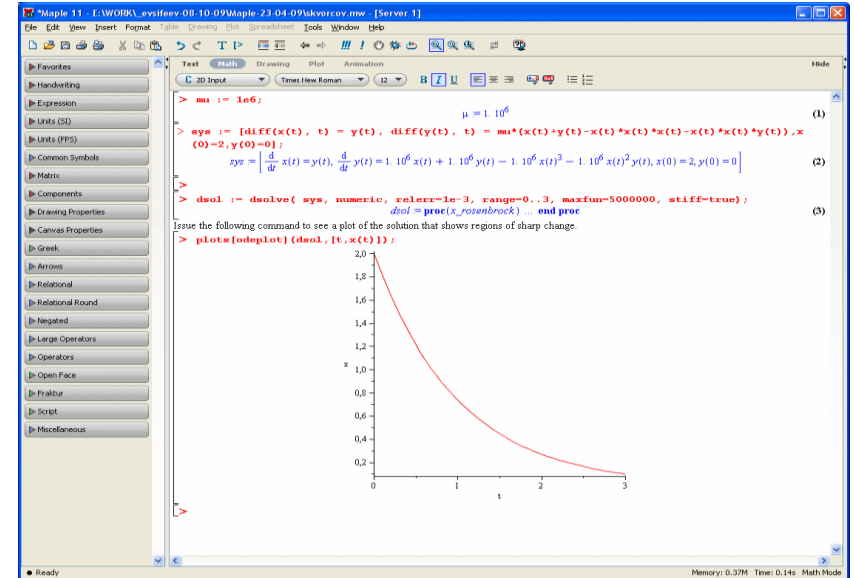
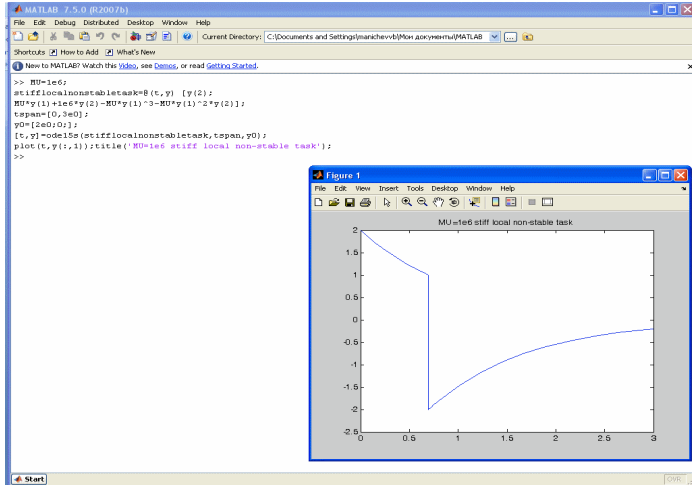
$$dy_1 / dt = y_2$$

$$dy_2 / dt = MU * (1 - y_1^2) * (y_1 + y_2)$$

$$y_1(0) = 2, y_2(0) = 0, t \in (0, 3)$$

Сравнение программ проведено для часто встречающихся на практике значений жесткости $MU=10^6$.

ОПИСАНИЯ, НЕКОРРЕКТНЫЕ РЕШЕНИЯ ТЕСТА 3 в MATLAB, Maple, C-Library NAG и КОРРЕКТНОЕ РЕШЕНИЕ в SADEL



ТЕСТ 4 Нелинейная жесткая система ОДУ для моделирования процессов реального лазера

$$dy_1 / dt = -y_1 * (\alpha * y_2 + \beta) + \gamma$$

$$dy_2 / dt = y_2 * (p * y_1 - \sigma) + \tau * (1 + y_1)$$

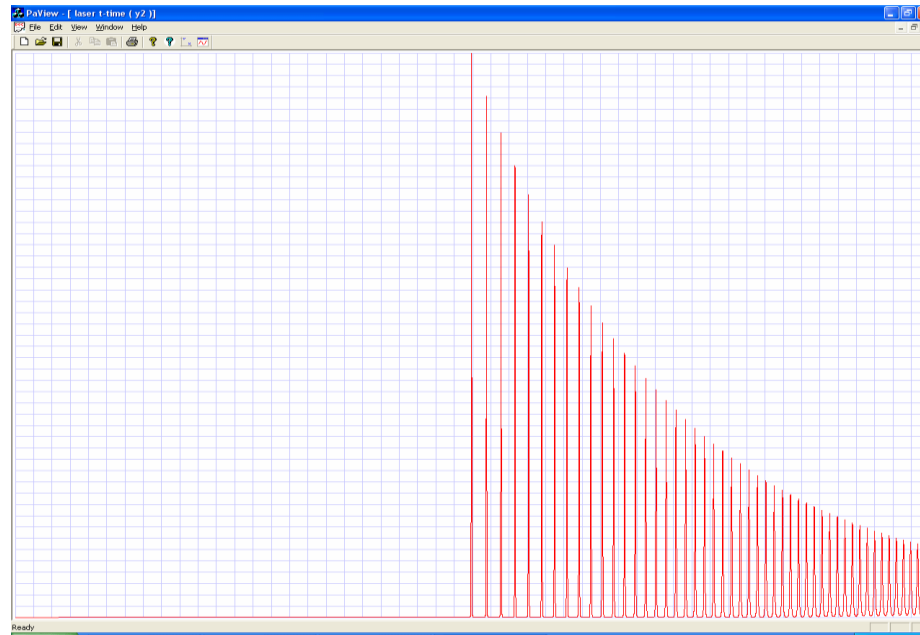
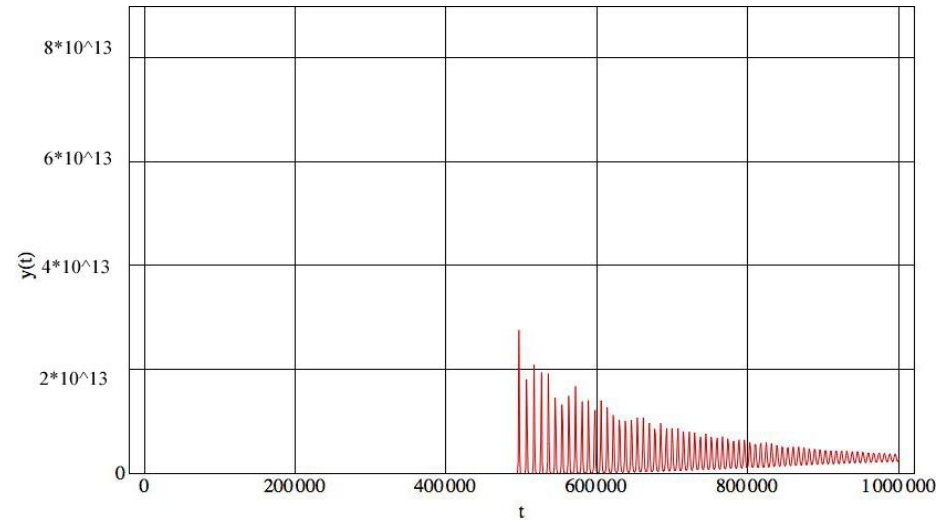
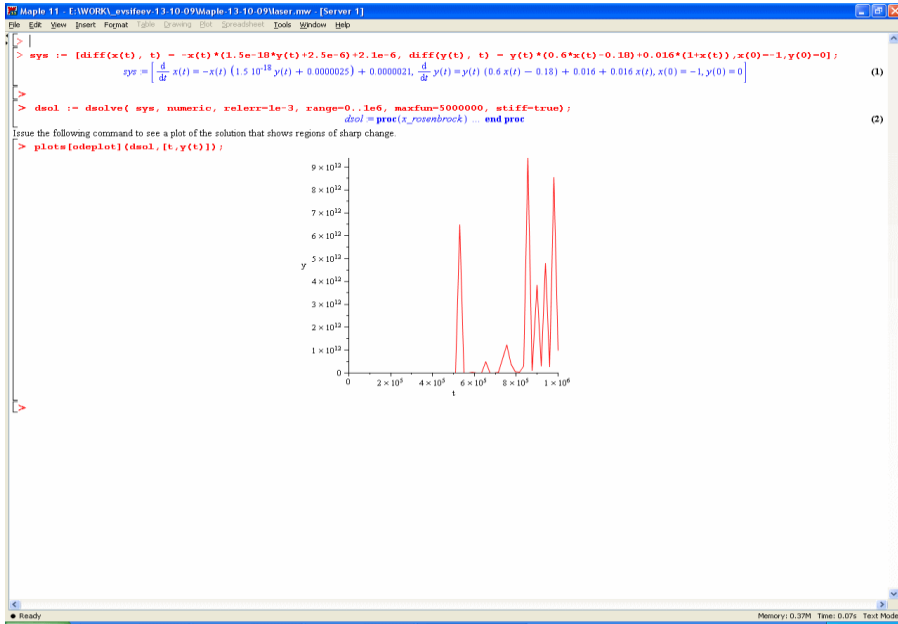
$$y_1(0) = -1, y_2(0) = 0, t \in (0, 10^6),$$

$$\alpha = 1.5 * 10^{-18}, \beta = 2.5 * 10^{-6}, \gamma = 2.1 * 10^{-6},$$

$$p = 0.6, \sigma = 0.18, \tau = 0.016$$

Сравнение программ проведено для параметров реального работающего лазера

Описание, НЕКОРРЕКТНОЕ РЕШЕНИЕ ТЕСТА 4 в Maple и C-Library NAG, КОРРЕКТНОЕ РЕШЕНИЕ в SADEL



Задача химической кинетики – реакция пиролиз (описание задачи в пакете Mathematica 8.0)

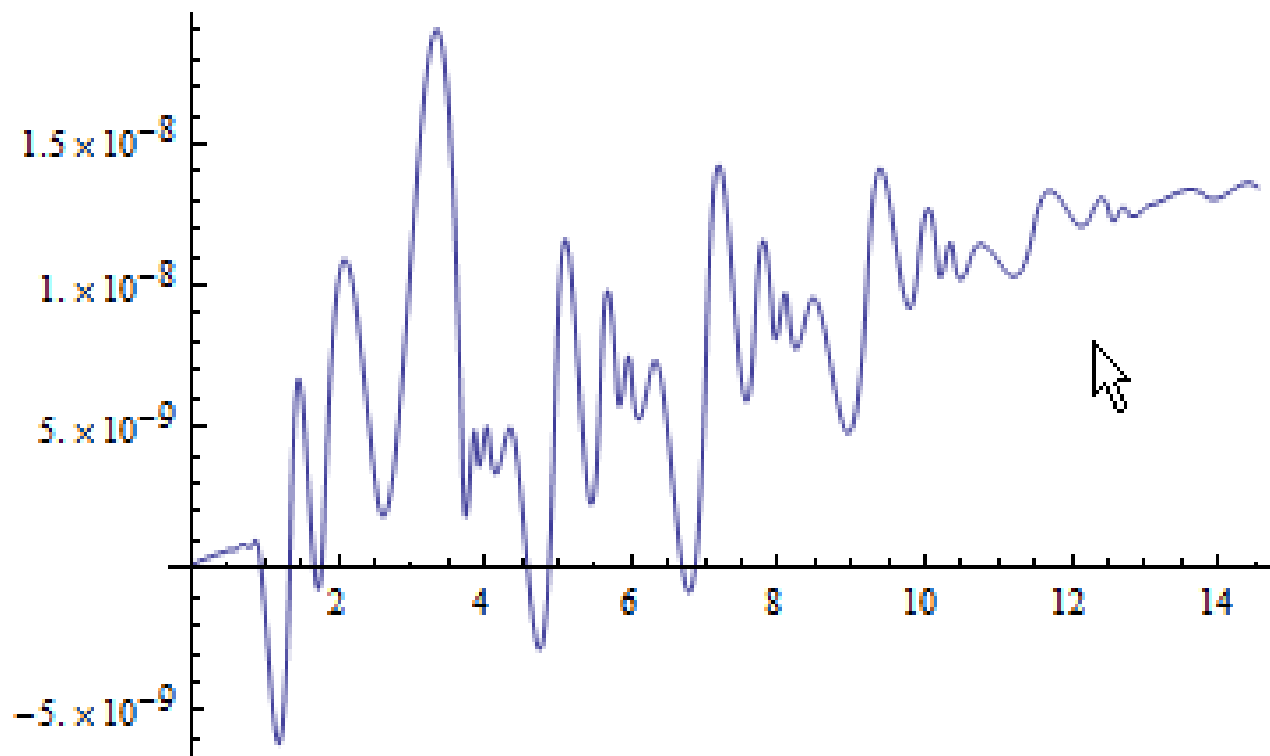
$k_1 = 0.001553;$ $k_2 = 24\,415\,734.935103;$ $k_3 = 41\,502.793067;$ $k_4 = 813\,084\,797.816872;$
 $k_5 = 9\,143\,838\,074.117590;$ $k_6 = 1\,499\,208\,614.309620;$ $k_7 = 5\,523\,511.416796;$ $k_8 = 3\,640\,227\,503.686290;$
 $k_9 = 907\,807.424730;$ $k_{10} = 5\,907\,025.708907;$ $k_{11} = 6\,064\,721\,569.931821;$ $k_{12} = 10\,000\,000\,000.000000;$
 \square $k_{13} = 0.030312;$ $k_{14} = 130.236748;$ \square

```
s = NDSolve[
{x1'(t) = -k1 x1(t) - k14 x1(t) x12(t) - k2 x1(t) x2(t) - k4 x1(t) x6(t) + k8 x4(t) x4(t),
 x2'(t) = 2 k1 x1(t) - k10 x2(t) x5(t) - k11 x2(t) x9(t) + k14 x1(t) x12(t) - k2 x1(t) x2(t) - k6 x2(t) x5(t) + k7 x8(t),
 x3'(t) = k10 x2(t) x5(t) + k11 x2(t) x9(t) + k2 x1(t) x2(t),
 x4'(t) = k2 x1(t) x2(t) - k3 x4(t) + k4 x1(t) x6(t) + k5 x5(t) x6(t) - 2 k8 x4(t) x4(t) + k9 x5(t) x8(t),
 x5'(t) = -k10 x2(t) x5(t) - k13 x5(t) + k3 x4(t) - k5 x5(t) x6(t) - k6 x2(t) x5(t) + k7 x8(t) + k8 x4(t) x4(t) - k9 x5(t) x8(t),
 x6'(t) = -k12 x6(t) x9(t) + k3 x4(t) - k4 x1(t) x6(t) - k5 x5(t) x6(t),
 x7'(t) = k12 x6(t) x9(t) + k4 x1(t) x6(t),
 x8'(t) = k14 x1(t) x12(t) + k6 x2(t) x5(t) - k7 x8(t) - k9 x5(t) x8(t),
 x9'(t) = k10 x2(t) x5(t) - k11 x2(t) x9(t) - k12 x6(t) x9(t),
 x10'(t) = k11 x2(t) x9(t) + k12 x6(t) x9(t),
 x11'(t) = k9 x5(t) x8(t),
 x12'(t) = k13 x5(t) - k14 x1(t) x12(t),
 x1(0) = 0.040746269, x2(0) = 0, x3(0) = 0, x4(0) = 0, x5(0) = 0.003955224, x6(0) = 0, x7(0) = 0, x8(0) = 0, x9(0) = 0, x10(0) = 0,
 x11(0) = 0, x12(0) = 0}, {x1, x2, x3, x4, x5, x6, x7, x8, x9, x10, x11, x12}, {t, 0, 14.55}, Method -> "StiffnessSwitching"]
```

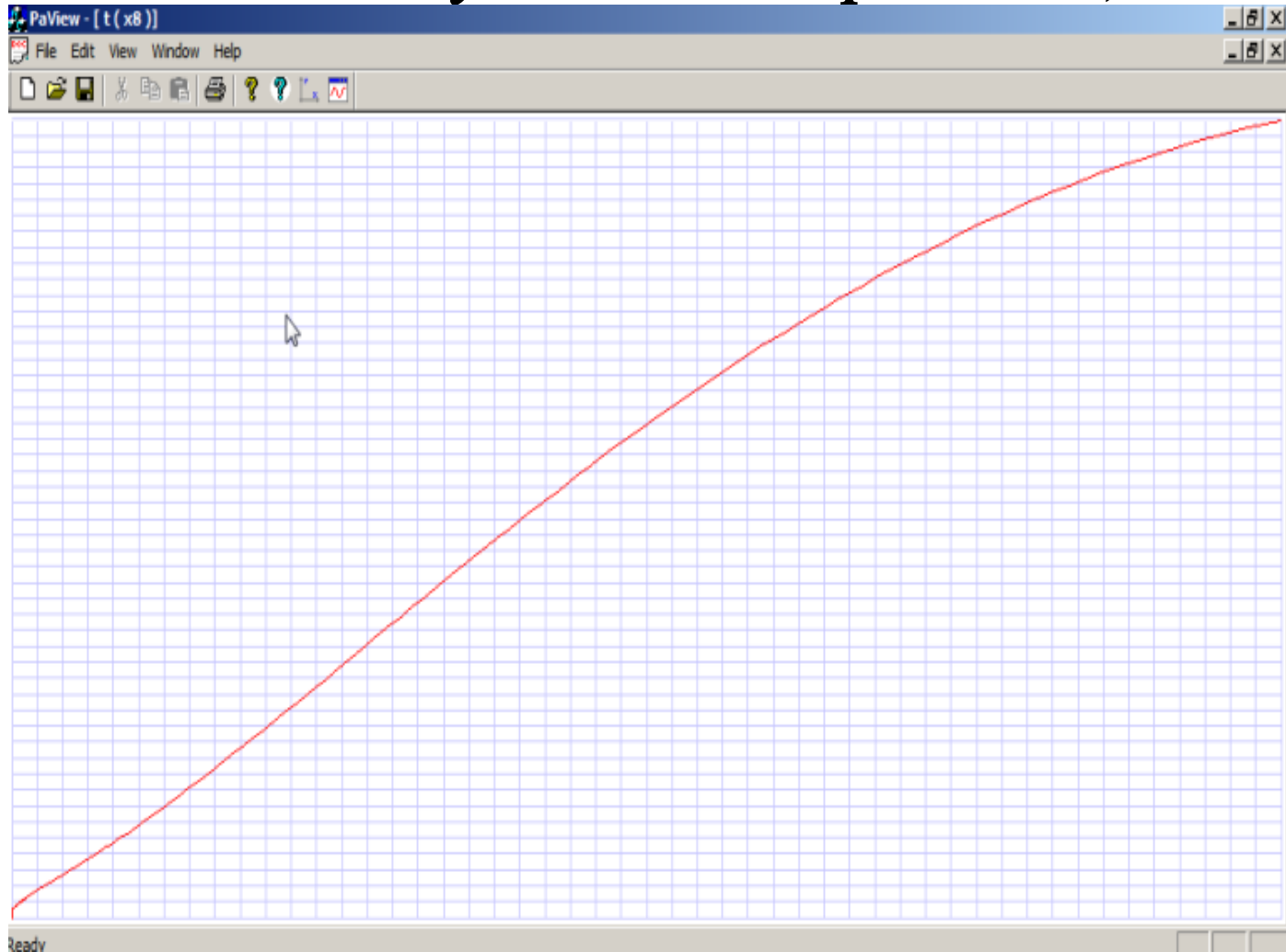
```
Plot[Evaluate[{x8[t]} /. s], {t, 0, 14.55}]
```

```
x8[14.55] /. s
```

Задача химической кинетики – реакция пиролиз (неверный график для переменной x_8 в пакете Mathematica 8.0 с параметрами по умолчанию)

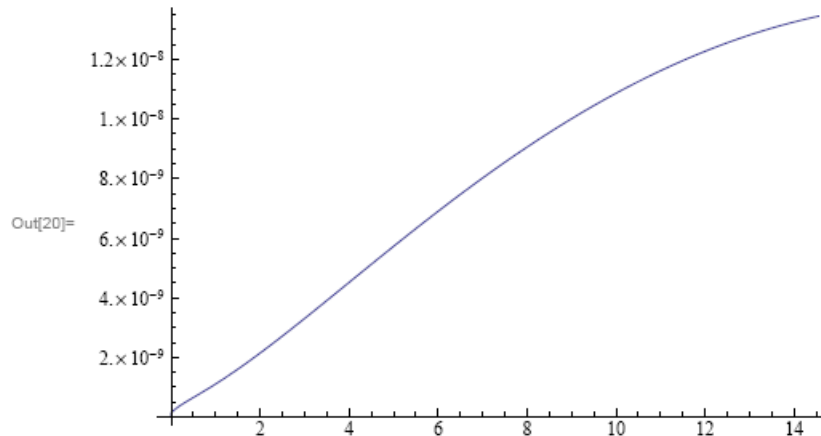


**Задача химической кинетики – реакция пиролиз
(верный график для переменной x_8 в *SADEL* для
точности по умолчанию $eps=0.001$)**



Задача химической кинетики – реакция пиролиз (исправленное описание и параметры интегрирования профессиональным математиком- программистом и верный график для переменной x_8 в пакете Mathematica 8.0)

```
In[19]:= s = NDSolve[system, varlist, {t, 0, Rationalize[14.55]},  
Method -> {"StiffnessSwitching", Method -> {"ExplicitRungeKutta", Automatic}},  
WorkingPrecision -> 32,  
MaxSteps -> Infinity];  
Plot[Evaluate[{x8[t]} /. s], {t, 0, Rationalize[14.55]}]
```



ОСНОВНЫЕ РЕЗУЛЬТАТЫ

1. Разработаны новые подходы, методы, алгоритмы и программы для достоверного решения систем ОДУ-ДАУ **в расширяемом координатном базисе зависимых переменных** на основе **$A(\pi/2)$ -устойчивых методов** достоверного численного интегрирования систем ДАУ при невысоких требованиях к точности интегрирования
2. Разработан **новый класс DA β c** численных методов решения систем ОДУ-ДАУ на основе систем алгебраических уравнений численного интегрирования, получены параметры для неявных **$A(\pi/2)$ -устойчивых методов** 2,4 и 6 порядков точности как развитие **Ab T c** методов Бутчера

Направление дальнейших исследований

1. Разработка новых высокоэффективных неявных ***DAbc*** методов интегрирования с полностью заполненными матрицами D и A в новых формулах интегрирования
2. Разработка новых методов и алгоритмов **автоматического выбора шага для $A(\pi/2)$ -устойчивых** неявных методов интегрирования
3. Разработка новых алгоритмов численно-аналитического **почленного вычисления Якобиана**
4. Разработка эффективных алгоритмов и программ решения **систем ЛАУ высокой и сверхвысокой размерности** с повышенной точностью вычислений
5. Разработка алгоритмов и программ **с повышенной точностью вычислений (extra precision calculation)** и с возможностью аппаратной реализации соответствующих алгоритмов для суперкомпьютеров