

УДК 519.711.3+551.5.001.57+517.957

Ю. Г. Игнатьев¹, А. Р. Самигуллина²

ЧИСЛЕННО-АНАЛИТИЧЕСКИЕ МЕТОДЫ МАТЕМАТИЧЕСКОГО МОДЕЛИРОВАНИЯ НЕЛИНЕЙНЫХ ДИНАМИЧЕСКИХ СИСТЕМ В СКМ MAPLE. II. АВТОМАТИЗАЦИЯ МАТЕМАТИЧЕСКОГО АНАЛИЗА НЕЛИНЕЙНЫХ МАТЕМАТИЧЕСКИХ МОДЕЛЕЙ

Описаны инструменты численно-аналитического моделирования нелинейных динамических систем на основе авторского прикладного пакета программ DifEqTools, приведены примеры вычислений.

Ключевые слова: нелинейные динамические системы, математическое моделирование, качественная теория дифференциальных уравнений, сплайновая аппроксимация, численные методы интегрирования, визуализация вычислений, прикладные математические пакеты.

PACS: 02.60.Cb, 07.05.Tr, 04.25.-g

This work was founded by the subsidy allocated to Kazan Federal University for the state assignment in the sphere of scientific activities.

1. Основные задачи визуализации математических моделей нелинейных динамических систем в системе компьютерной математики Maple

Важнейшим этапом численно-аналитического моделирования нелинейных динамических систем является визуализация численных результатов компьютерного моделирования. Визуализация математической модели позволяет не только выявить основные закономерности описываемого ею процесса, но и обнаружить некоторые детали, недоступные выявлению аналитическими методами исследования. В этой статье мы будем рассматривать инструменты исследования так называемых *обобщенно - механических* систем [2], описываемых системой обыкновенных нелинейных дифференциальных уравнений [3]:

$$y_i^{(n_i)} = F_i(y_1, \dots, y_N, y_1', \dots, y_N', y_1'', \dots, y_N'', \dots, y_1^{(n_1-1)}, \dots, y_N^{(n_N-1)}, t); \quad (i = \overline{1, N}), \quad (1)$$

где $y^{(n)} = d^n y / dt^n$ — обозначение n -той производной функции $y(t)$ по независимой переменной t , временной переменной³, а F_i — непрерывно-дифференцируемые функции своих переменных. Будем в дальнейшем полагать выполненными начальные условия для системы (1):

$$y_i^{(k)}(t) \Big|_{t=t_0} = C_i^k; \quad (k = \overline{1, n_i - 1}; i = \overline{1, N}), \quad (2)$$

соответствующие задаче Коши, где C_i^k — начальные значения производных k -го порядка функций $y_i(t)$. Ясно, что путем известных преобразований, которые автоматически выполняет представленный ранее пакет программ DifEq [4]⁴, система (2) преобразуется к стандартной нормальной системе (вообще говоря, неавтономной) обыкновенных дифференциальных уравнений (ОДУ)

$$\frac{dx^i}{dt} = f^i(x^1, \dots, x^n, t) \quad (i = \overline{1, n}) \Rightarrow \dot{\mathbf{r}} = \mathbf{f}(t). \quad (3)$$

с начальными условиями Коши

$$(x^1(t_0), \dots, x^n(t_0)) = (x_{(0)}^1, \dots, x_{(0)}^n) \Rightarrow \mathbf{r}(t_0) = \mathbf{r}_{(0)}. \quad (4)$$

¹Е-mail: ignatev_yu@rambler.ru

²Е-mail: alsu_sam@rambler.ru

³Хотя это может быть и не обязательно физическое время.

⁴как и расширенный пакет DifEqTools

Поэтому все функции $y_i(t)$ и их производные до $n_i - 1$ порядка включительно мы будем называть *динамическими переменными*:

$$y_1(t), \dots, y_N(t), y_1'(t), \dots, y_N'(t), y_1''(t), \dots, y_N''(t), \dots, y_1^{(n_1-1)}(t), \dots, y_N^{(n_N-1)}(t).$$

При этом необходимо оперативно получить графическую информацию об исследуемой динамической системе в различных форматах:

1. графики зависимости одной из динамических переменных от временной переменной, $x^i(t)$ на заданном интервале $t \in [t_1, t_2]$;
2. графики двумерных фазовых траекторий вида $[x^i(t), x^k(t)]$ на заданном интервале временной переменной $t \in [t_1, t_2]$;
3. трехмерные графики фазовой траектории вида $[x^i(t), x^k(t), x^l(t)]$ на заданном интервале временной переменной $t \in [t_1, t_2]$;
4. графики зависимости заданной функции динамических переменных от временной переменной, $f(x^1(t), \dots, x^N(t))$ на заданном интервале $t \in [t_1, t_2]$;
5. графики зависимости интегралов и производных от заданных функций динамических переменных от временной переменной, на заданном интервале $t \in [t_1, t_2]$;
6. графики двумерных фазовых траекторий двух заданных функций динамических переменных от временной переменной, $[f_1(x^1(t), \dots, x^N(t)), f_2(x^1(t), \dots, x^N(t))]$, на заданном интервале временной переменной $t \in [t_1, t_2]$;
7. графики трехмерных фазовых траекторий трех заданных функций динамических переменных от временной переменной, $[f_1(x^1(t), \dots, x^N(t)), f_2(x^1(t), \dots, x^N(t)), f_3(x^1(t), \dots, x^N(t))]$, $[x^i(t), x^k(t)]$ на заданном интервале временной переменной $t \in [t_1, t_2]$;
8. динамические графики (анимации) для всех указанных случаев.

При этом следует также предусмотреть возможность изменять методы решения и вывода численных и численно - аналитических решений, разработанных в предыдущих авторских пакетах **DifEq** и **SDplines**:

1. возможность изменять методы численного решения системы нелинейных дифференциальных уравнений (1) с начальными условиями (2);
2. расширенные возможности управления точностью решений и допустимым числом итерационных шагов;
3. возможность представления решения в управляемом сплайновом формате.

Целью этой статьи, как и ряда предыдущих и последующих, является создание программного инструментария автоматизации численно-аналитического моделирования и полного исследования нелинейных динамических систем произвольного порядка. Центральной идеей Авторы является освобождение исследователей от громоздких вычислений и сложных специальных процедур программирования, позволив, тем самым, исследователю сосредоточиться на основных задачах научного исследования. Ясно, что процесс совершенствования такой системы программ не имеет конца, поэтому мы периодически будем вносить коррективы в программные продукты, сопровождая их необходимыми комментариями и примерами.

В этой статье мы опишем авторский расширенный пакет программ **DifEqTools** численно - аналитического моделирования нелинейных динамических систем и автоматизированного построения графических компьютерных моделей⁵ и приведем примеры исследований на основе этого пакета. Пакет вместе

⁵Пакет обновлен 05.05.2017

с примерами использования размещен в свободном доступе на сайте журнала <http://www.stfi.ru/ru/software.html>. Авторы просят исследователей ссылаться на этот пакет при его использовании.

2. Инструменты исследования математических моделей нелинейных динамических систем в системе компьютерной математики Maple

2.1. Формат ввода задачи Коши и вывода информации о системе ОДУ

Пусть задача Коши определяется системой (1) и начальными условиями (2). Система обыкновенных дифференциальных уравнений (разрешенных относительно старших производных), как и система начальных условий, записывается в пакете `DiffEqTools` в форме списка (внутри квадратных скобок) даже в случае одиночного уравнения. Для простоты приведем пример системы формата записи двух линейных дифференциальных уравнений максимального третьего порядка, записанного с помощью дифференциального оператора $(D@@n)(x)(t) \equiv \frac{d^n x(t)}{dt^n}$:

```
>Eqs1 := [(D@@3)(x)(t) = -0.1*D(y)(t) - x(t), (D@@2)(y)(t) = 0.1*D(x)(t) - y(t)];
```

В обычных обозначениях `convert(Eqs1, diff)` эта система имеет вид;

$$\left[\frac{d^3 x(t)}{dt^3} = -0.1 \frac{d(t)}{dt} - x(t), \frac{d^2 y(t)}{dt^2} = 0.1 \frac{dx(t)}{dt} - y(t) \right]; \quad (5)$$

Введем также начальные условия для системы (5):

```
>Ics1 := [x(0)=2, D(x)(0)=0, (D@@2)(x)(0)=0, y(0)=0, D(y)(0)=1];
```

$$\Rightarrow \left[x(0) = 2, \left. \frac{dx}{dt} \right|_{t=0} = 0, \left. \frac{d^2 x}{dt^2} \right|_{t=0} = 0, y(0) = 0, \left. \frac{dy}{dt} \right|_{t=0} = 1 \right] \quad (6)$$

Заметим, что полная информация о введенной задаче Коши извлекается с помощью команды

```
>DiffEqTools[SysCauchy_ConvNorm](Eqs1, Ics1);
```

В результате получим:

```
[5, 3], [[X[1](t), X[2](t)], [Y[1](t), Y[2](t)], [Z[1](t)]];
[x(t) = X[1](t), (D(x))(t) = Y[1](t), ((D@@2)(x))(t) = Z[1](t),
y(t) = X[2](t), (D(y))(t) = Y[2](t), t = t],
[diff(X[1](t), t) = Y[1](t), diff(Y[1](t), t) = Z[1](t), diff(X[2](t), t) = Y[2](t),
diff(Z[1](t), t) = .1*Y[2](t)-X[1](t), diff(Y[2](t), t) = .1*Y[1](t)-X[2](t)],
[X[1](0) = 2, Y[1](0) = 0, Z[1](0) = 0, X[2](0) = 0, Y[2](0) = 1], 0]
```

В полученном списке первый список `[5, 3]` содержит число динамических переменных (5) и наивысший порядок производных системы ОДУ (3). Второй список содержит перечисление некоторых канонических переменных задачи, причем имя переменной (X, Y) относится к номеру искомым функций по порядку правой части уравнений, а индекс — к порядку производной этой функции. Третий список содержит набор замен динамических переменных к каноническим переменным, в которых система ОДУ принимает нормальную форму (четвертый список). Пятый список представляет запись начальных условий в канонических переменных, а шестой список содержит один элемент — начальное значение временной переменной.

Обратим внимание также на команду пакета `DiffEqTools[InitialVars](Eqs)`, которая выводит список имен всех динамических переменных по заданной системе ОДУ. Для системы (5) исполнение команды

```
>DiffEqTools[InitialVars](Eqs1)
```

дает следующий список имен:

$$\left[x(t), y(t), t, \left[\frac{dx(t)}{dt}, \frac{dy(t)}{dt} \right], \left[\frac{d^2x(t)}{dt^2} \right] \right] \quad (7)$$

Здесь первый список содержит имена всех искомых функций, а также имя независимой переменной; второй список содержит имена первых производных искомых функций, а третий список в нашем случае содержит лишь одну величину – имя второй производной функции $x(t)$, так как высший порядок производной этой функции в системе ОДУ равен 3.

2.2. Вывод численных решений

Пакет `DifEqTools` позволяет автоматически выводить численные решения задачи Коши (1) – (2) в двух форматах. Первый из них реализуется командой

```
DifEqTools[NumDsolve](Eqs, ICS, [Method]),
```

где `Eqs` – система ОДУ и `ICS` – система начальных условий, записанных в виде списка (пример см. выше); `[Method]` – список параметров численных методов интегрирования системы ОДУ. Этот список должен содержать хотя бы один параметр – метод интегрирования, в форме `method=Method`, где `Method` – конкретный метод интегрирования системы ОДУ в СКМ Maple (см. Табл. 1).

Табл. 1. Некоторые методы численного интегрирования систем ОДУ в СКМ Maple

Название метода	Имя параметра в Maple	Характеристики метода
метод Рунге - Кутты - Фелберга 4-5 порядка	rkf45	Наиболее простой и быстрый метод интегрирования
метод Рунге - Кутты 7-8 порядка	dverk78	Метод интегрирования повышенной точности
метод Розенброка	rosenbrock	Метод интегрирования, сравнимый по точности с rkf45, применимый для интегрирования жестких систем ОДУ
классический метод	classical	Метод интегрирования с фиксированным размером шага интегрирования, дает большие ошибки
метод Тейлора	taylorseries	Метод интегрирования, основанный на разложении в ряд Тейлора, низкая скорость вычисления

Заметим, что метод Розенброка тоже с успехом работает для жестких систем и, пожалуй, является самым быстрым при сохранении точности вычислений. Кроме метода численного интегрирования в списке `[Method]` могут быть указаны дополнительные параметры численного интегрирования: абсолютная точность вычисления `abserr`, относительная допустимая ошибка вычисления `relerr`, максимально допустимое число шагов интегрирования `maxfun`. Заметим, что еще в версии Maple 18 максимальное значение этого параметра составляло 500000. В последней версии Maple 2016 это ограничение снято. Во-всяком случае интегрирование осуществляется даже с `maxfun=15000000`.

Командная процедура `DifEqTools[NumDsolve](Eqs, ICS, [Method])` выводит решение в формате списка. Приведем пример численного решения задачи Коши (5) – (6):

```
>SS:=DifEqTools[NumDsolve](Eqs1, ICS1, [method=rkf45]);
```

```
SS:=[[proc(t)...end proc,proc(t)...end proc],[proc(t)...end proc,proc(t)...end proc],[proc(t)...end proc]]
```

```
SS(0);SS(1);
```

```
[[2., 0.], [0., 1.], [0.1000000000000000]]
[[1.70287092152896, .848308700994169], [-.932817872564193, .568714400167878], [-1.91440840979363]]
```

Аналогичная командная процедура `NumericDsolve(Eqs, IC, [Method])` выводит решение в формате `operator`:

```
>RR:=DifEqTools[NumericDsolve](Eqs1, ICS1, [method=rkf45]);
```

```
RR:=[t=proc(t)...end proc,x=proc(t)...end proc,D(x)=proc(t)...end proc,
D^(2)(x)=proc(t)...end proc,y=proc(t)...end proc,D(y)=proc(t)...end proc]
```

```
>RR(0);RR(1);
```

```
[t = 0., x(0) = 2., (D(x))(0) = 0., ((D@@2)(x))(0) = .1000000000000000, y(0) = 0., (D(y))(0) = 1.]
[t = 1., x(1) = 1.70287092152896, (D(x))(1) = -.932817872564193, ((D@@2)(x))(1) =
-1.91440840979363, y(1) = .848308700994169, (D(y))(1) = .568714400167878]
```

Этот формат удобен для осуществления подстановки решения в пользовательские функции.

Команда `DifEqTools[NumericDsolve](Eqs, IC, [method=name])` выводит численное решение задачи Коши для системы ОДЕ `Eqs`, разрешенных относительно старших производных, начальных условий `IC` с помощью метода численного интегрирования `name`, при этом имя метода численного интегрирования может быть дополнено специальными опциями интегрирования, абсолютной и относительной точностью, максимальным числом шагов и другими.

Продемонстрируем применение команды к решению уравнений свободных колебаний⁶

```
>Eq_osc:=[diff(x(t),t$2)=-x(t)];
```

$$Eq_osc := \left[\frac{d^2 x(t)}{dt^2} = -x(t) \right] \quad (8)$$

```
>IC_osc:=[x(0)=0,D(x)(0)=1];
```

$$[x(0) = 0, D(x)(0) = 1] \quad (9)$$

Точное решение этой задачи Коши есть: $x(t) = \sin(t)$. Применяя команду, введем процедуру вывода решения:

```
PP1:=DifEqTools[NumericDsolve](Eq_osc, IC_osc, [method=rkf45]);
```

```
PP1 := [t=proc(t) ... end proc,x=proc(t) ... end proc,D(x)=proc(t) ... end proc]
```

Само численное решение можно вывести простой функцией:

```
>PP1(1);
```

```
[t = 1., x(1) = .841471101155307, (D(x))(1) = .540302331778567];
```

Рассмотрим еще один пример:

```
> Eqs:=[(D@@2)(x)(t)=-0.1*D(y)(t)-x(t), (D@@2)(y)(t)=-0.1*D(x)(t)-y(t)];
```

$$Eqs := \left[D^{(2)}(x)(t) = -0.1D(y)(t) - x(t), D^{(2)}(y)(t) = -0.1D(x)(t) - y(t) \right]; \quad (10)$$

Введем также начальные условия для системы уравнений (10):

```
>ICS:=[x(0)=2,D(x)(0)=1,y(0)=0,D(y)(0)=0];
```

$$ICS := [x(0) = 2, D(x)(0) = 1, y(0) = 0, D(y)(0) = 0]; \quad (11)$$

⁶Читателя пусть не смущает простота приведенного примера, численные методы пакета Maple с одинаковой скоростью интегрируют линейные и нелинейные системы ОДУ. Данный пример приведен лишь в силу его простоты и прозрачности, а также для возможности сравнения с точным решением задачи Коши. Более сложные примеры будут рассмотрены ниже.

2.3. Программа вывода произвольных функций от численного решения задачи Коши

Команда `NumericDsolveFun(Eqs, ICs, [Metod], tau, F(t))` устроены аналогично предыдущим, но теперь она уже позволяют выводить функции $F(t)$ от численных решений. Здесь `tau` - значение временной переменной, при котором выводится функция $F(t)$, а $F(t)$ может быть и списком выводимых функций (а также и их производных) вида:

$$F(t) = [f_1(x(t), y(t), \dots, t), \dots, f_k(x(t), y(t), \dots, t)].$$

Приведем пример для задачи Коши (5) – (6):

```
>RRF:=(tau)->DifEqTools[NumericDsolveFun](Eqs1,ICS1,method=rkf45,
tau,[sin(x(t))^2,y(t)^2,D(x)(t)^2]);
```

```
RRF := proc → DifEqToolsNumericDsolveFun(Eqs1,ICS1,method = rkf45,
tau,[sin(x(t))^2 + : 2,y(t)^2,(D(x))(t)^2]) end proc
```

```
>RRF(1);
```

```
[1.982657493, .719627652182414, .870149183375188]
```

2.4. Программа вычисления определенных интегралов от функций, построенных на численных решениях системы ОДУ

Определенные интегралы вида:

$$J(s_0, s) = \int_{s_0}^s f(x(t), y(t), \dots, t) dt, \quad (12)$$

где $x(t), y(t)$ – численные решения задачи Коши для системы ОДУ в пакете `DifEqTools`, а $f(x(t), y(t), \dots, t)$ – заданная функция этих решений, вычисляются с помощью команды `NumericDsolveIntFun(Eqs, IC, metod, ff, L, N, k)`,

где первые три параметра описаны выше, $ff(t) = [t, f(x(t), y(t), \dots, t)]$, $L = [s_0, s]$, N – число отрезков разбиения интервала интегрирования, k – метод интегрирования ($k=T$ – метод трапеций, $k=S$ – метод Симпсона).

Приведем примеры интегрирования. Рассмотрим интеграл вида

$$J(s) = \int_0^s e^{x(t)} dt, \quad (13)$$

где $x(t)$ – численное решение задачи Коши (8) – (9). Этот интеграл вычисляется методом трапеций с разбиением интервала интегрирования на 100 отрезков командой

```
>SS_T:=(s)->DifEqTools[NumericDsolveIntFun](Eq_osc,IC_osc,
method=rkf45,[t,exp(x(t))],[0,s],100,T);
```

```
>SS_T(1);
```

```
1.6551289004143868
```

при интегрировании методом трапеций и командой

```
>SS_S:=(s)->DifEqTools[NumericDsolveIntFun](Eq_osc,IC_osc,
method=rkf45,[t,exp(x(t))],[0,s],100,S);
```

```
>SS_S(1);
```

```
1.63186967646492
```

при интегрировании методом Симпсона.

На рисунках (Рис. 1 и Рис. 2) показаны результаты численного интегрирования на основе интегрирования численного решения задачи Коши (8) – (9) методом трапеций и Симпсона, соответственно.

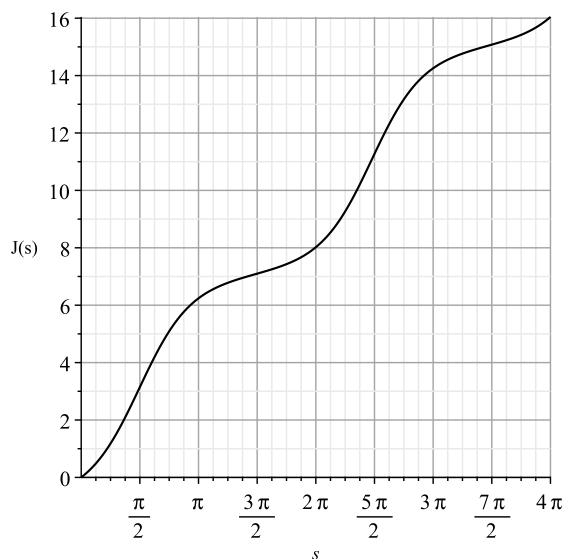


Рис. 1. Интеграл $J(s)$, вычисленный на решении задачи Коши методом трапеций.

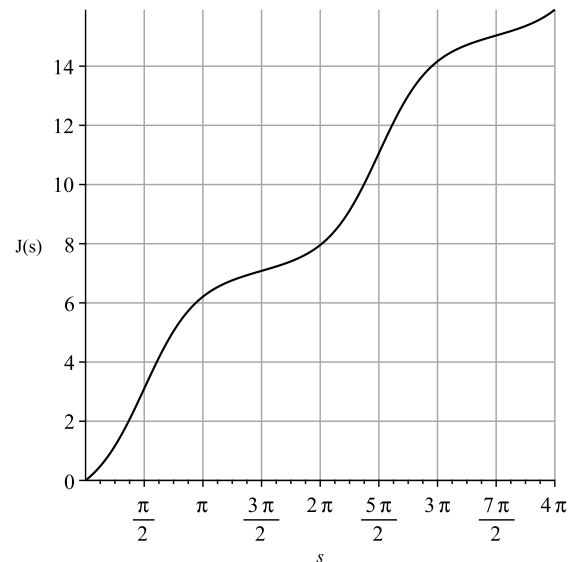


Рис. 2. Интеграл $J(s)$, вычисленный на решении задачи Коши методом Симпсона.

Для сравнения точности и скорости интегрирования вычислим интегралы на точном решении задачи Коши (8) – (9) и на основе команд численного интегрирования. Рассмотрим, например, интегралы:

$$S2(s) = \int_0^s x(s)^2 ds \equiv \frac{1}{2}s - \frac{1}{4} \sin 2s;$$

$$C2(s) = \int_0^s x'(s)^2 ds \equiv \frac{1}{2}s + \frac{1}{4} \sin 2s, \quad (14)$$

так что:

$$S2(s) + C2(s) \equiv s. \quad (15)$$

Вычислим интегралы $S2(s)$ и $C2(s)$ методом трапеций при разбиении интервала интегрирования на 1000 отрезков:

```
>SS_2T:=(s)->DifEqTools[NumericDsolveIntFun](Eq_osc,IC_osc,method=rkf45,
[t,x(t)^2],[0,s],1000,T);
CC_2T:=(s)->DifEqTools[NumericDsolveIntFun](Eq_osc,IC_osc,method=rkf45,
[t,D(x)(t)^2],[0,s],1000,T);
CS2T:=(s)->SS_2T(s)+CC_2T(s);
```

и методом Симпсона с таким же разбиением

```
>SS_2S:=(s)->DifEqTools[NumericDsolveIntFun](Eq_osc,IC_osc,method=rkf45,
[t,x(t)^2],[0,s],1000,S);
CC_2S:=(s)->DifEqTools[NumericDsolveIntFun](Eq_osc,IC_osc,method=rkf45,
[t,D(x)(t)^2],[0,s],1000,S);
CS2S:=(s)->SS_2S(s)+CC_2S(s);
```

и полученные результаты сравним с точным результатом (15), выводя с помощью простой процедуры и время вычислений, T:

```
>T:=time():
plot(CS2T(s)-s,s=0..2*Pi);
time()-T;
```

```
>T:=time():
plot(CS2S(s)-s,s=0..2*Pi);
time()-T;
```

На рисунках представлены результаты интегрирования.

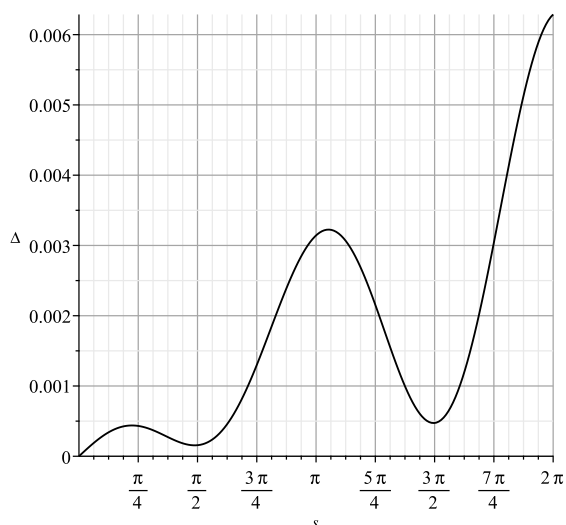


Рис. 3. Абсолютная ошибка интегрирования функции численного решения методом трапеций при разбиении интервала на 1000 отрезков. Время вычислений 177 с.

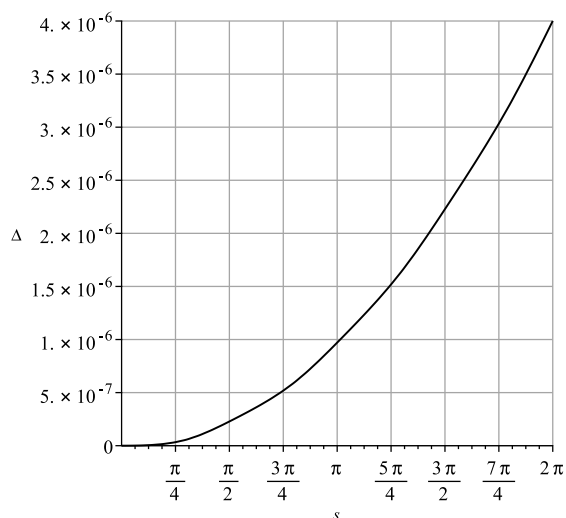


Рис. 4. Абсолютная ошибка интегрирования функции численного решения методом Симпсона при разбиении интервала на 1000 отрезков. Время вычислений 60 с.

Таким образом, метод Симпсона дает точность примерно на 3 порядка выше, чем метод трапеций с одновременным уменьшением времени вычислений примерно в 3 раза.

2.5. Программа вычисления производных от функций, построенных на численных решениях системы ОДУ

Хотя производные от численных решений и их функции можно выводить и с помощью описанной выше команды `NumericDsolveFun(Eqs,ICs,[Metod],tau,F(t))`, в пакете `DifEqTools` существует и специальная команда `Differential(Eqs,IC,[metod],[dff])` прямого вычисления производных с заданной точностью `d`, где первые три параметра описаны выше, а список `[dff]` определяется следующим образом:

`[dff]=[k,θ,n]`, где `k` — номер выводимой функции, `n` — порядок ее производной.

Покажем пример применения команды для вычисления первой производной численного решения рассмотренной задачи Коши

```
>PP3:=(t)->DifEqTools[NumericDifferential](Eq_osc,IC_osc,[method=rkf45],[1,θ,1,t,θ.001]);
```

и второй производной

```
>PP4:=(t)->DifEqTools[NumericDifferential](Eq_osc,IC_osc,[method=rkf45],[1,θ,2,t,θ.001]);
```

с точностью `d=θ.001`.


```
>PP3(Pi/3);
      0.5000000612220347

>PP4(Pi/3);
     -0.866025493890739
```

На рисунке (Рис. 5) показан график второй производной численного решения задачи Коши (8) – (9), выведенный с точностью $d=0.01$. Видно, что график совпадает с графиком функции $-\sin(t)$.

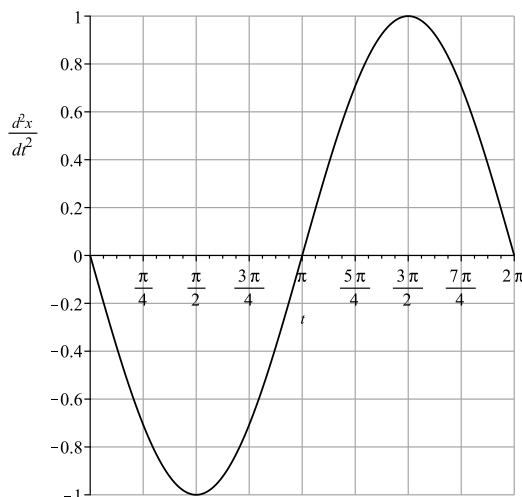


Рис. 5. График второй производной численного решения задачи Коши (8) – (9), точность вычисления производной – 0.01.

3. Благодарности

Авторы благодарны ректорату Казанского университета и дирекции Института математики и механики им. Н.И. Лобачевского за материальную поддержку экспериментального проекта необходимым компьютерным и другим оборудованием, а также лицензионным программным обеспечением.

ЛИТЕРАТУРА

1. Игнатьев Ю.Г., Самигуллина А.Р. Численно - аналитические методы математического моделирования нелинейных динамических систем в СКМ Maple // Пространство, время и фундаментальные взаимодействия. 2016. Вып. 4(17). С. 78–105.
2. Игнатьев Ю.Г. Математическое моделирование фундаментальных объектов и явлений в системе компьютерной математики Maple. Лекции для школы по математическому моделированию. Казань: Казанский университет, 2014. 298 с.
3. Игнатьев Ю.Г., Самигуллина А.Р. Программный комплекс численно - аналитического моделирования нелинейных динамических систем в СКМ Maple // Пространство, время и фундаментальные взаимодействия. 2016. Вып. 4(17). С. 147–148.
4. Игнатьев Ю.Г., Самигуллина А.Р. Программный комплекс численно-аналитического моделирования нелинейных динамических систем в СКМ Maple // Пакет программ. URL: <http://www.stfi.ru/ru/software.html>.

Поступила в редакцию 02.07.2017

Игнатьев Юрий Геннадьевич, д. ф.-м. н., профессор, кафедра высшей математики и математического моделирования, Институт математики и механики им. Н.И. Лобачевского, Казанский федеральный университет, 420008, Россия, г. Казань, ул. Кремлевская, 35.

E-mail: ignatev_yu@rambler.ru

Самигуллина Алсу Ринатовна, ассистент, кафедра высшей математики и математического моделирования, Институт математики и механики им. Н.И. Лобачевского, Казанский федеральный университет, 420008, Россия, г. Казань, ул. Кремлевская, 35.

E-mail: alsu_sam@rambler.ru

Просьба ссылаться на эту статью следующим образом:

Игнат'ев Ю.Г., Самигуллина А.Р. Численно-аналитические методы математического моделирования нелинейных динамических систем в СКМ Maple. II. Автоматизация математического анализа нелинейных математических моделей // Пространство, время и фундаментальные взаимодействия. 2017. № 2. С. 74–83.

Yu. G. Ignat'ev, A. R. Samigullina

Numerically - analytical methods for mathematical modeling of nonlinear dynamic systems in CAS Maple. II. Automation of the mathematical analysis of nonlinear mathematical models

Keywords: nonlinear dynamical systems, mathematical modeling, qualitative theory of differential equations, spline approximation, numerical integration methods, visualization computing, application mathematical packages.

PACS: 02.60.Cb, 07.05.Tp, 04.25.-g

Describe the tools of numerical and analytical modeling of nonlinear dynamic systems based on the author's applied software package DifEqTools and examples of calculations.

REFERENCES

1. Ignat'ev Yu.G., Samigullina A.R. Numerically-analytical methods for mathematical modeling of nonlinear dynamic systems in CAS Maple, *Space, Time and Fund. Interact.*, 2016, issue 4(17), p. 78–105.
2. Ignatyev Yu.G. *Matematicheskoe modelirovanie fundamental'nyh objektov i yavlenij v sisteme komp'yuternoj matematiki Maple. Lekcii dlya shkoly po matematicheskomu modelirovaniyu* (Mathematical modeling of fundamental objects and phenomena in system of computer mathematics Maple. Lectures for school on mathematical modeling), Kazan: Kazan University, 2013, 298 p.
3. Ignat'ev Yu.G., Samigullina A.R. Program complex for numerical and analytical mathematical modeling of nonlinear dynamic systems in CAS Maple, *Space, Time and Fund. Interact.*, 2016, issue 4(17), p. 147–148.
4. Ignat'ev Yu.G., Samigullina A.R. Program complex for numerical and analytical mathematical modeling of nonlinear dynamic systems in CAS Maple, *Space, Time and Fund. Interact.*, 2016. <http://www.stfi.ru/ru/software.html>.

Received 02.07.2017

Ignat'ev Yuri Gennadievich, Doctor of Physics and Mathematics, Professor, Lobachevsky Institut of Mathematics and Mechanics, Kazan Federal University, ul. Kremlyovskaya, 35A, Kazan, 420008, Russia.

E-mail: ignatev_yu@rambler.ru

Samigullina Alsu Rinatovna, assistant lecturer, Lobachevsky Institute of Mathematics and Mechanics, Kazan Federal University, ul. Kremlyovskaya, 35, Kazan, 420008, Russia.

E-mail: alsu_sam@rambler.ru

Please cite this article in English as:

Ignat'ev Yu. G., Samigullina A. R. Numerically - analytical methods for mathematical modeling of nonlinear dynamic systems in CAS Maple. II. Automation of the mathematical analysis of nonlinear mathematical models, *Space, Time and Fund. Interact.*, 2017, no. 2, pp. 74–83.