Глава 9

Решение дифференциальных уравнений

9.1. Введение в решение дифференциальных уравнений 698
9.2. Примеры решения дифференциальных уравнений 705
9.3. Специальные средства решения дифференциальных уравнений 710
9.4. Инструментальный пакет решения дифференциальных уравнений DEtools 716
9.5. Графическая визуализация решений дифференциальных уравнений 722
9.6. Углубленный анализ дифференциальных уравнений 732
9.7. Решение дифференциальных уравнений с частными производными 735
9.8. Сложные колебания в нелинейных системах и средах 740
9.9. Интерактивное решение дифференциальных уравнений в Maple 745
9.10. Анализ линейных функциональных систем 748
9.11. Решение дифференциальных уравнений СКМ Mathematica 751
9.12. Численное решение ДУ в системе Mathcad 756
9.13. Символьное решение ДУ в системе Mathcad 773
9.14. О реализации в Mathcad вариационных методов 779
9.15. Математическое моделирование в Mathcad колебательных систем 781
9.16. Моделирование в Mathcad биологических и экономических систем 790
9.17. Новые возможности в решении дифференциальных уравнений в Mathematica 794

Дифференциальные уравнения лежат в основе математического моделирования различных, в том числе физических, систем и устройств [1, 126–137]. Решению таких уравнений посвящена эта глава. В ней рассмотрено как аналитическое, так и численное решение дифференциальных уравнений различного вида – линейных и нелинейных, классических и специальных, например в частных производных и с учетом двухсторонних граничных условий. Описание сопровождается множеством наглядных примеров, реализованных в различных СКМ. Примеры, приведенные в этой главе для системы Maple 11, пригодны без ограничений для предшествующих версий Maple 9/9.5/10 и для недавно появившейся новой версии Maple 12.

9.1. Введение в решение дифференциальных уравнений

9.1.1. Дифференциальные уравнения первого порядка

Дифференциальные уравнения (*ДУ*) – это уравнения, при записи которых встречаются производные того или иного порядка. Простейшее *ДУ первого порядка*

$$y' = \frac{dy}{dx} = f(x,y) \tag{9.1}$$

в общем случае имеет решение в виде зависимости $y(x)$, которое зависит от вида функции $f(x, y)$ и начальных значений x_0 и y_0. Это решение может быть аналитическим, конечно-разностным или численным.

В качестве примера аналитического решения ДУ первого порядка в среде СКМ Maple 9.5 запишем *ДУ радиоактивного распада атомов* (N – число атомов в момент времени t, $g = 1/c$):

```
> restart: deq:=diff(N(t),t)=-g*N(t);
```

$$deq := \frac{d}{dt}\mathrm{N}(t) = -g\mathrm{N}(t)$$

Используя функцию `dsolve`, которая более подробно будет описана чуть позже, получим его общее аналитическое решение:

```
> dsolve(deq,N(t));
```

$$N(t) = _C1\,\mathbf{e}^{(-gt)}$$

В решении присутствует *произвольная постоянная* _$C1$. Но ее можно заменить на постоянную $\mathrm{N}(0) = \mathrm{N}_0$, означающую начальное число атомов в момент $t = 0$:

```
> dsolve({deq,N(0)=No},N(t));
```

$$N(t) = No\,\mathbf{e}^{(-gt)}$$

Если конкретно N_0=100 и g=4, то получим:

```
> No := 100;g:=3;
```

$$No := 100$$
$$\mathrm{g} := 3$$

```
> s:=dsolve({deq,N(0)=No},N(t));assign(s);
```

$$s := \mathrm{N}(t) = 100\mathbf{e}^{(-3t)}$$

Теперь мы можем воспользоваться полученной зависимостью $N(t)$ и построить ее график:

```
> plot(N(t),t=0..3,color=black);
```

Этот график, показанный на рис. 9.1 (вместе с представленными выше вычислениями), описывает хорошо известный апериодический экспоненциальный закон уменьшения числа атомов вещества в ходе его радиоактивного распада.

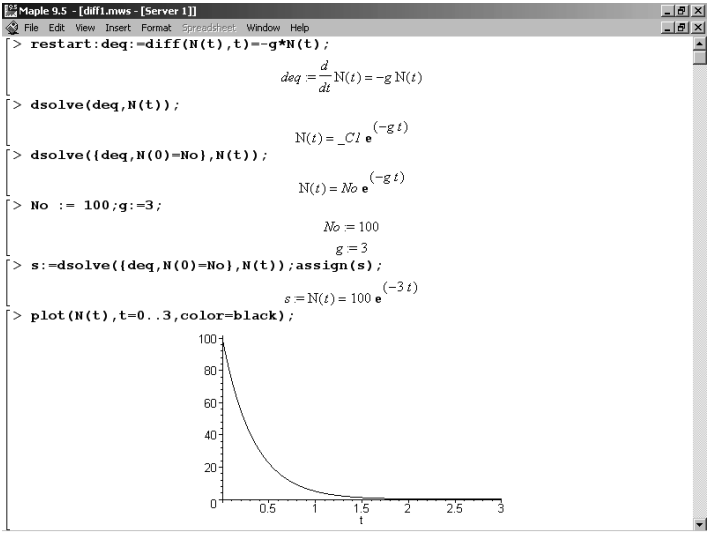


Рис. 9.1. Пример решения дифференциального уравнения радиоактивного распада

Подобные зависимости, кстати, характерны для напряжения на конденсаторе C при его разряде через резистор R, для тока в LR-цепи и для многих простых физических явлений, описывающихся дифференциальным уравнением первого порядка.

9.1.2. Системы дифференциальных уравнений

Встроенные в математические системы функции обычно решают *систему* из обыкновенных дифференциальных уравнений (ОДУ), представленную в *форме Коши*:

$$\begin{cases} y_1(x_0) = y_{0,1} \\ y_2(x_0) = y_{0,2} \\ \ldots\ldots\ldots\ldots \\ y_n(x_0) = y_{0,n} \end{cases}, \quad \begin{cases} y_1' = f_1(x, y_1, y_2, ..., y_n) \\ y_2' = f_2(x, y_1, y_2, ..., y_n) \\ \ldots\ldots\ldots\ldots\ldots\ldots \\ y_n' = f_n(x, y_1, y_2, ..., y_n) \end{cases}.$$

Здесь левая система задает начальные условия, а вторая представляет систему ОДУ.

9.1.3. Сведение ДУ высокого порядка к системам ОДУ первого порядка

Часто встречаются *ДУ высокого порядка*:

$y^{(n)} = f(x, y, y', y'', \ldots y^{(n-1)})$,

где

$y(x_0) = y_0, y'(x_0) = y_{0,1}, y''(x_0) = y_{0,2}, \ldots, y^{(n-1)}(x_0) = y_{0,n-1}$.

Обозначив

$y_1(x) = y(x),\ y_2(x) = y'(x), \ldots, y_n(x) = y^{(n-1)}(x)$

и

$y_{0,0} = y(x_0), y_{0,1} = y'(x_0), \ldots, y_{0,n-1} = y^{(n-1)}(x_0)$,

теперь решение этого уравнения можно свести к решению *системы ОДУ*:

$$\begin{cases} y_1(x_0) = y_{0,0} \\ y_2(x_0) = y_{0,1} \\ \ldots\ldots\ldots\ldots \\ y_n(x_0) = y_{0,n-1} \end{cases} \quad \begin{cases} y_1' = y_2 \\ y_2' = y_3 \\ \ldots\ldots\ldots\ldots \\ y_{n-1}' = y_n \\ y_n' = f(x, y_1, y_2, \ldots, y_{n-1}) \end{cases}$$

В таком виде ДУ n-го порядка может решаться стандартными средствами решения систем ОДУ, входящими в большинство математических систем.

9.1.4. Решение задачи на полет камня

В качестве примера аналитического решения системы дифференциальных уравнений рассмотрим постановку типичной задачи моделирования «Бросок камня», позволяющую описать полет камня-точки с массой m, брошенного под углом α_0 к горизонту. Пренебрегая сопротивлением воздуха и зная начальные координаты камня x_0 и y_0 и его начальную скорость v_0, получим следующую модель в виде дифференциальных уравнений:

$$m \cdot \frac{d^2x}{dt^2} = 0, \quad m \cdot \frac{d^2y}{dt^2} = -m \cdot g, \quad v_x = \frac{dx}{dt}, \quad v_y = \frac{dy}{dt} \tag{9.2}$$

при следующих начальных условиях:

$x(0) = x_0, \quad y(0) = y_0, \quad v_x(0) = v_0 \cdot \cos\alpha_0, \quad v_y(0) = v_0 \cdot \sin\alpha_0.$

Надо найти зависимости $x(t), y(t), v_x(t), v_y(t)$.

Решение этой задачи есть в любом учебнике физики. Тем не менее выполним его. Из (9.2) запишем систему ОДУ первого порядка:

$$\frac{dv_x}{dt} = 0, \quad \frac{dv_y}{dt} = -g, \quad v_x = \frac{dx}{dt}, \quad v_y = \frac{dy}{dt}. \tag{9.3}$$

После интегрирования получим:

$$v_x(t) = C_1, \quad v_y(t) = C_2 - g \cdot t, \quad x(t) = C_3 + C_1 \cdot t, \quad y(t) = C_4 + C_2 \cdot t - \frac{g \cdot t^2}{2}. \tag{9.4}$$

Определив константы интегрирования из начальных условий, окончательно запишем:

$$x(t) = x_0 + v_0 \cdot \cos a_0 \cdot t \qquad y(t) = y_0 + v_0 \cdot \sin a_0 \cdot t - \frac{g \cdot t^2}{2}$$

$$v_x(t) = v_0 \cdot \cos a_0 \qquad v_y(t) = v_0 \cdot \sin \alpha_0 - g \cdot t$$

Из аналитического решения вытекает, что полет камня при отсутствии сопротивления воздуха происходит строго по параболической траектории, причем она на участках полета камня вверх и вниз симметрична.

9.1.5. Классификация дифференциальных уравнений в Maple

Дифференциальные уравнения могут быть самого разного вида. На рис. 9.2 представлен раздел справки Maple 9.5 с классификацией дифференциальных уравнений. В ней представлены:

- 20 дифференциальных уравнений первого порядка;
- 25 дифференциальных уравнений второго порядка;
- 6 типов дифференциальных уравнений высшего порядка;
- основные функции решения дифференциальных уравнений.

В качестве примера работы с классификатором выберем решение дифференциального уравнения Бернулли. Для этого активизируем на рис. 9.2 гиперссылку с его именем – Bernoulli. Появится окно справки по этому уравнению, показанное на рис. 9.3 с открытой позицией меню **Edit**.

С помощью команды **Copy Examples** в позиции **Edit** меню можно перенести примеры решения с окна справки в буфер Clipboard операционной системы Windows. После этого командой **Paste** в меню **Edit** окна документа можно пере-

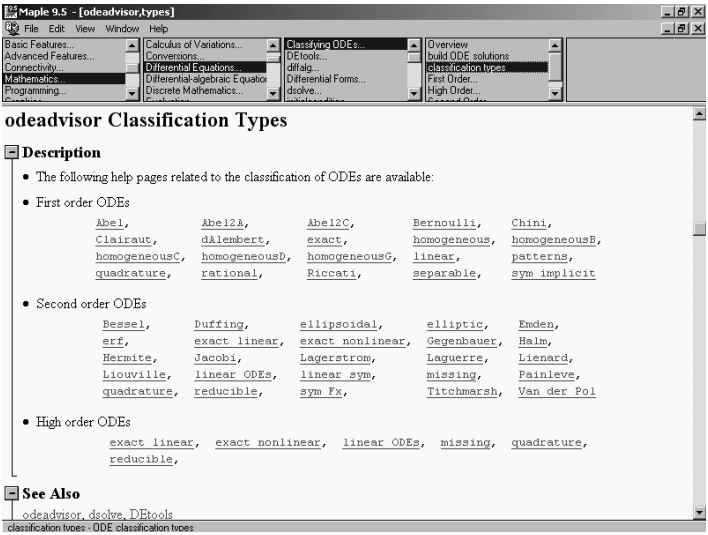


Рис. 9.2. Классификация дифференциальных уравнений

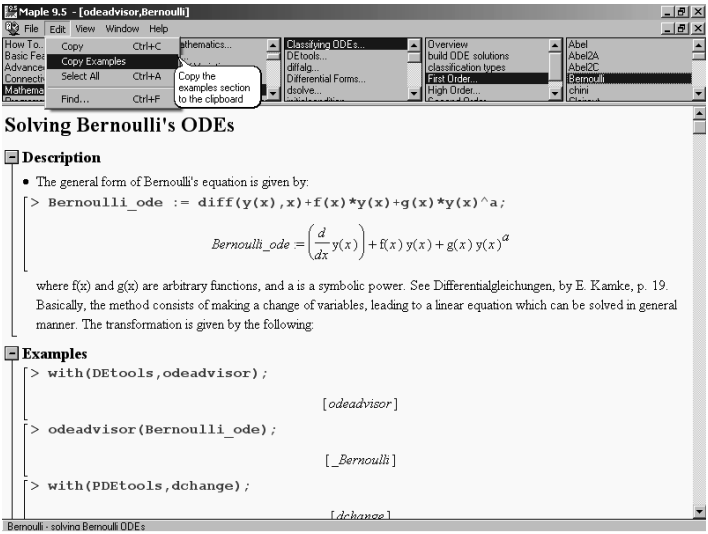


Рис. 9.3. Окно справки по решению дифференциального уравнения Бернулли

нести примеры в текущий документ – желательно (но не обязательно) новый. Теперь можно наблюдать решение выбранного дифференциального уравнения – рис. 9.4.

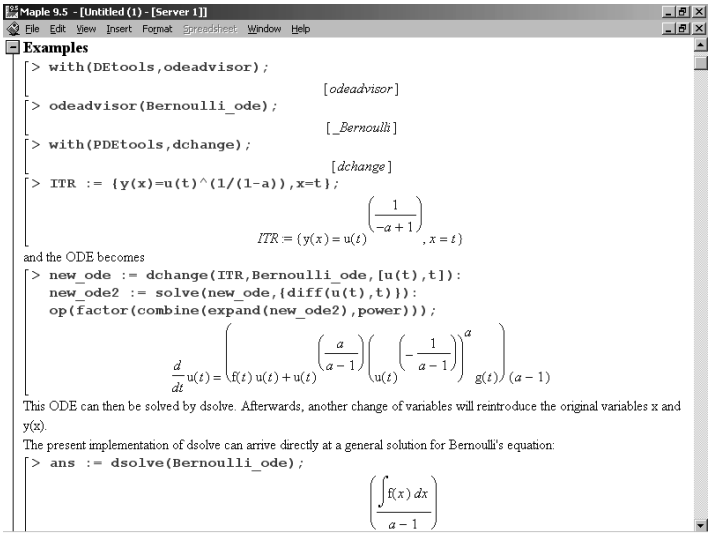


Рис. 9.4. Пример решения дифференциального уравнения Бернулли из справки

Уже в Maple 9.5 и в последующих версиях средства решения дифференциальных уравнений подверглись переработке. В систему введены новые методы решения для дифференциальных уравнений Абеля, Риккати и Матье и инициализации и решения уравнений с кусочными функциями, улучшены алгоритмы решения численными методами. Детальное описание этих новинок можно найти в справке по разделу What's New....

9.1.6. Функция решения ДУ dsolve

Maple позволяет решать одиночные дифференциальные уравнения и системы дифференциальных уравнений как аналитически, так и в численном виде. Для решения системы простых дифференциальных уравнений (*задача Коши*) используется функция `dsolve` в разных формах записи:

```
dsolve(ODE)
dsolve(ODE, y(x), extra_args)
dsolve({ODE, ICs}, y(x), extra_args)
dsolve({sysODE, ICs}, {funcs}, extra_args)
```

Здесь ODE – одно обыкновенное дифференциальное уравнение или система из дифференциальных уравнений первого порядка с указанием начальных условий, y(x) – функция одной переменной, Ics – выражение, задающее начальные условия, {sysODE} – множество дифференциальных уравнений, {funcs} – множество неопределенных функций, extra_argument – опция, задающая тип решения.

Параметр extra_argument задает класс решаемых уравнений. Отметим основные значения этого параметра:

- exact – аналитическое решение (принято по умолчанию);
- explicit – решение в явном виде;
- system – решение системы дифференциальных уравнений;
- ICs – решение системы дифференциальных уравнений с заданными начальными условиями;
- formal series – решение в форме степенного многочлена;
- integral transform – решение на основе интегральных преобразований Лапласа, Фурье и др.;
- series – решение в виде ряда с порядком, указываемым значением переменной Order;
- numeric – решение в численном виде.

Возможны и другие опции, подробное описание которых выходит за рамки данной книги. Его можно найти в справке по этой функции, вызываемой командой ?dsolve.

Для решения задачи Коши в параметры dsolve надо включать начальные условия, а при решении краевых задач – краевые условия. Если Maple способна найти решение при числе начальных или краевых условий меньше порядка системы, то в решении будут появляться неопределенные константы вида _C1, _C2 и т. д. Они же могут быть при аналитическом решении системы, когда начальные условия не заданы. Если решение найдено в неявном виде, то в нем появится параметр _T. По умолчанию функция dsolve автоматически выбирает наиболее подходящий метод решения дифференциальных уравнений. Однако в параметрах функции dsolve в квадратных скобках можно указать предпочтительный метод решения дифференциальных уравнений. Допустимы следующие методы:

```
> 'dsolve/methods'[1];
```

[*quadrature*, *linear*, *Bernoulli*, *separable*, *inverse_linear*, *homogeneous*, *Chini*, *lin_sym*, *exact*, *Abel*, *pot_sym*]

Более полную информацию о каждом методе можно получить, используя команду ?dsolve,method и указав в ней конкретный метод. Например, команда ?dsolve,linear вызовет появление страницы справочной системы с подробным описанием линейного метода решения дифференциальных уравнений.

9.1.7. Уровни решения дифференциальных уравнений в Maple

Решение дифференциальных уравнений может сопровождаться различными комментариями. Команда

```
infolevel[dsolve] := n:,
```

где n – целое число от 0 до 5, управляет детальностью вывода. По умолчанию задано n = 0. Значение n = 5 дает максимально детальный вывод.

Производные при записи дифференциальных уравнений могут задаваться функцией `diff` или оператором дифференцирования `D`. Выражение `sysODE` должно иметь структуру множества и содержать, помимо самой системы уравнений, их начальные условия.

Читателю, всерьез интересующемуся проблематикой решения линейных дифференциальных уравнений, стоит внимательно просмотреть разделы справки по ним и ознакомиться с демонстрационным файлом linearoade.mws, содержащим примеры решения таких уравнений в закрытой форме.

9.2. Примеры решения дифференциальных уравнений

9.2.1. Примеры аналитического решения ОДУ первого порядка

Отвлекшись от физики, приведем несколько примеров на составление и решение дифференциальных уравнений первого порядка в аналитическом виде:

```
> dsolve(diff(y(x),x)-a*x=0,y(x));
```

$$y(x)=\frac{ax^2}{2}+_C1$$

```
> dsolve(diff(y(x),x)-y(x)=exp(-x),y(x));
```

$$y(x)=\left(-\frac{1}{2}e^{(-2x)}+_C1\right)e^{x}$$

```
> dsolve(diff(y(x),x)-y(x)=sin(x)*x,y(x));
```

$$y(x)=-\frac{1}{2}\cos(x)x-\frac{1}{2}\cos(x)-\frac{1}{2}\sin(x)x+e^{x}\,_C1$$

```
> infolevel[dsolve] := 3:
> dsolve(diff(y(x),x)-y(x)=sin(x)*x,y(x));
```

```
Methods for first order ODEs:
- Trying classification methods --
trying a quadrature
trying 1st order linear
<- 1st order linear successful
```

$$y(x) = -\frac{1}{2}\cos(x)x - \frac{1}{2}\cos(x) - \frac{1}{2}\sin(x)x + \mathbf{e}^{x}_C1$$

Обратите внимание на вывод в последнем примере. Он дан при уровне вывода n=3.

Следующие примеры иллюстрируют возможность решения одного и того же дифференциального уравнения `ode_L` разными методами:

```
> ode_L := sin(x)*diff(y(x),x)-cos(x)*y(x)=0;
```

$$ode_L := \sin(x)\left(\frac{\partial}{\partial x}y(x)\right) - \cos(x)y(x) = 0$$

```
> dsolve(ode_L, [linear], useInt);
```

$$y(x) = _C1\mathbf{e}^{\left(\int \frac{\cos(x)}{\sin(x)}dx\right)}$$

```
> value(%);
```

$$y(x) = _C1\sin(x)$$

```
> dsolve(ode_L, [separable], useInt);
```

$$\int \frac{\cos(x)}{\sin(x)}dx - \int^{y(x)} \frac{1}{_a}d_a + _C1 = 0$$

```
> value(%);
```

$$\ln(\sin(x)) - \ln(y(x)) + _C1 = 0$$

```
> mu := intfactor(ode_L);
```

$$\mu := \mathrm{int\,factor}\left(\sin(x)\left(\frac{\partial}{\partial x}y(x)\right) - \cos(x)y(x) = 0\right)$$

```
> dsolve( mu*ode_L, [exact], useInt);
```

$$y(x) = \frac{_C1}{\frac{1}{2}\tan\left(\frac{1}{2}x\right) + \frac{\frac{1}{2}}{\tan\left(\frac{1}{2}x\right)}}$$

Разумеется, приведенными примерами далеко не исчерпываются возможности аналитического решения дифференциальных уравнений.

9.2.2. Полет тела, брошенного вверх

Из приведенных выше примеров видно, что для задания производной используется ранее рассмотренная функция `diff`. С помощью символа `$` в ней можно задать производную более высокого порядка.