Обыкновенные дифференциальные уравнения в Maple

Для решения обыкновенных дифференциальных уравнений (ОДУ) в Maple имеется команда *dsolve*(), которая пытается найти общее решение в аналитическом виде, и пакет *DEtools* с возможностями численного решения задачи Коши и мощной графикой. Кроме того, Maple позволяет эффективно получить приближенное аналитическое решение с помощью подходящих рядов. Для решения дифференциальных уравнений в частных производных можно обратиться к пакету *PDEtools*.

Аналитическое решение ОДУ и их систем.

Для нахождения в Maple общего решения в символьном виде можно применить команду *dsolve*() в следующем формате: *dsolve*(*DEq*, *options*), где *DEq* – ОДУ, *options* – дополнительные (необязательные) параметры. Они могут указывать метод решения задачи. По умолчанию ищется аналитическое решение, для чего может быть использован аргумент команды *type*=*exact*. При составлении дифференциальных уравнений для обозначения производной применяется команда *diff*:

```
> dsolve(diff(y(x), x) = x^2·y(x))
```

$$y(x) = \_C1\, e^{\frac{1}{3}x^3}$$

Общее решение ОДУ зависит от произвольных постоянных, число которых равно порядку дифференциального уравнения. В Maple такие константы обозначаются как *_C1*, *_C2* и т.д.

Общее решение неоднородного линейного дифференциального уравнения выводится в виде суммы его частного решения (без произвольных постоянных) и общего решения соответствующего однородного дифференциального уравнения (с произвольными постоянными):

```
> de := diff(y(x), x$2) − 4·diff(y(x), x) + 3·y(x) = 2·exp(−3·x); dsolve(de, y(x));
```

$$de := \frac{d^2}{dx^2}y(x) - 4\left(\frac{d}{dx}y(x)\right) + 3\,y(x) = 2\,e^{-3x}$$

$$y(x) = e^{x}\,\_C2 + e^{3x}\,\_C1 + \frac{1}{12}\,e^{-3x}$$

Команда *dsolve* выдает решение дифференциального уравнения в невычисляемом формате. Для дальнейшей работы с решением можно отделить правую часть полученного равенства командой *rhs*(%). Например, построим график полученной функции при *_C1*=1, *_C2*=0:

$$y(x) = e^{3x}\_C2 + e^{x}\_C1 + \frac{1}{12}e^{-3x}$$

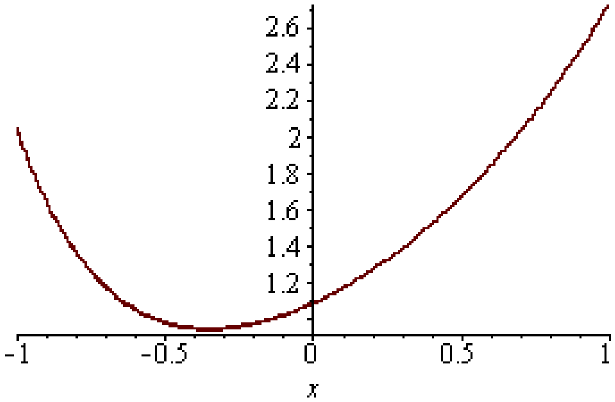
> $r := rhs(\%)$;*#формула общего решения*

$$r := e^{3x}\_C2 + e^{x}\_C1 + \frac{1}{12}e^{-3x}$$

> $y := subs(\_C1 = 1, \_C2 = 0, r)$;*#подстановка в него значений произвольных постоянных*

$$y := e^{x} + \frac{1}{12}e^{-3x}$$

> $plot(y, x = -1\,..1)$*#построение графика полученного частного решения*



Команда *dsolve* представляет возможность найти фундаментальную систему решений (базисные функции) ОДУ. Для этого в параметрах команды *dsolve* следует указать *output=basis*:

> $restart;\ de := diff(y(x), x\$2) - 4 \cdot diff(y(x), x) + 5 \cdot y(x) = 0;\ dsolve(de, y(x), output = basis)$

$$de := \frac{d^2}{dx^2}y(x) - 4\left(\frac{d}{dx}y(x)\right) + 5\,y(x) = 0$$

$$\left[e^{2x}\sin(x), e^{2x}\cos(x)\right]$$

Команда *dsolve* может найти решение задачи Коши, если помимо дифференциального уравнения задать начальные условия для искомой функции. Для обозначения производных в начальных условиях используется дифференциальный оператор D. Например, условие $y''(1) = 2$ следует записать в виде $D^{(2)}(y)(1) = 2$.

Команда *dsolve* может найти решение системы обыкновенных дифференциальных уравнений, если использовать следующий ее формат: *dsolve*($\{de_1, de_2, \ldots, de_n\}, \{x(t), \ldots, y(t)\}$), где $\{de_1, de_2, \ldots, de_n\}$ – множество уравнений, входящих в заданную систему, $x(t), \ldots, y(t)$ – неизвестные функции.

Найдем решение системы дифференциальных уравнений относительно двух неизвестных функций:

> $ds := diff(x(t), t) = x(t) - 2 \cdot y(t) + t,\ diff(y(t), t) = 2 \cdot x(t) - 3 \cdot y(t);$

$$ds := \frac{d}{dt}x(t) = x(t) - 2\,y(t) + t,\ \frac{d}{dt}y(t) = 2\,x(t) - 3\,y(t)$$

> $dsolve(\{ds\}, \{x(t), y(t)\});$

$$\left\{x(t) = e^{-t}\_C2 + e^{-t}t\_C1 + 3t - 5, y(t) = e^{-t}\_C2 + e^{-t}t\_C1 - \frac{1}{2}e^{-t}\_C1 - 4 + 2t\right\}$$

Найдены две функции $x(t)$ и $y(t)$, которые зависят от двух произвольных постоянных *_C1* и *_C2*.

Заметим, что переменные *x* и *y* заданной системы необходимо вводить как функции независимого аргумента *t*, то есть в виде $x(t)$ и $y(t)$ соответственно. Если этого не сделать, Maple выдаст ошибку:

```
> ds := diff(x(t),t) = x - 2·y + t, diff(y(t),t) = 2·x - 3·y;
```

$$ds := \frac{d}{dt}x(t) = x - 2y + t,\ \frac{d}{dt}y(t) = 2x - 3y$$

```
> dsolve({ds}, {x(t), y(t)});
Error, (in dsolve) ambiguous input: the variables {x, y} and the functions {x(t), y(t)} cannot both appear in the system
```

Приближенное решение ОДУ и их систем.

В случае, когда аналитическое решение дифференциального уравнения не может быть найдено, можно использовать разложение неизвестной функции в степенной ряд или численное приближение решения.

Чтобы найти приближенное решение дифференциального уравнения в виде степенного ряда, в команде *dsolve* следует после переменных указать параметр *type=series* (или просто *series*). Для того чтобы указать порядок *n* разложения, следует перед командой *dsolve* вставить его определение с помощью команды *Order:=n*. По умолчанию значение системной переменной *Order* равно 6.

Если ищется общее решение дифференциального уравнения в виде разложения в степенной ряд, то коэффициенты при степенях *x* найденного разложения будут содержать неизвестные значения $y(0)$ функции в нуле и ее производных $D(y)(0)$, $(D^{(2)})(y)(0)$ и т.д. Полученное в строке вывода выражение будет иметь вид, похожий на разложение искомого решения в ряд Маклорена, но с другими коэффициентами при степенях *x*. Для выделения частного решения следует задать начальные условия $y(0)=y0$, $D(y)(0)=y1$, $(D^{(2)})(y)(0)=y2$ и т.д., причем количество этих начальных условий должно совпадать с порядком соответствующего дифференциального уравнения.

Разложение в степенной ряд имеет тип *series*, поэтому напомним, что для дальнейшей работы с этим рядом его следует преобразовать в полином с помощью команды *convert*(%,*polynom*), а затем выделить правую часть полученного выражения командой *rhs*(%).

Найдем точное и приближенное в виде степенного ряда до 4-го порядка решения ОДУ $y''' - y' = x\cos x$. Сравним их при начальных условиях $y(0)=1, y'(0)=1, y''(0)=2$, построив соответствующие графики в одной системе координат. Используем следующий набор команд Maple:

> *restart*;

> *# Ввод исходных данных*
Order := 4; *de* := *diff*(*y*(*x*), *x*$3) − *diff*(*y*(*x*), *x*) = *x*·cos(*x*); *cond* := *y*(0) = 1, D(*y*)(0) = 1, D$^{(2)}$(*y*)(0) = 2

$$Order := 4$$

$$de := \frac{d^3}{dx^3} y(x) - \left(\frac{d}{dx} y(x)\right) = x\cos(x)$$

$$cond := y(0) = 1, D(y)(0) = 1, D^{(2)}(y)(0) = 2$$

> *# Аналитическое решениеdsolve*({*de*, *cond*}, *y*(*x*)) ; *y1* := *rhs*(%)

$$y(x) = \frac{1}{2} e^{-x} + \frac{3}{2} e^{x} - \cos(x) - \frac{1}{2} x\sin(x)$$

$$y1 := \frac{1}{2} e^{-x} + \frac{3}{2} e^{x} - \cos(x) - \frac{1}{2} x\sin(x)$$

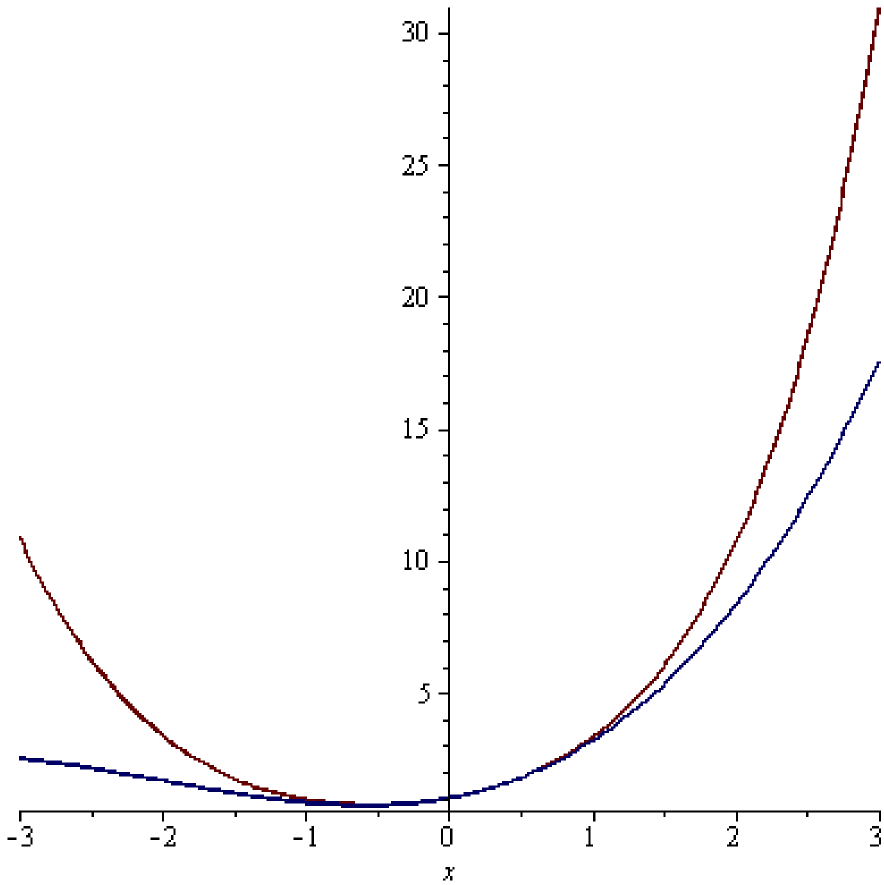
> *# Приближенное решение в виде ряда и его преобразование в полином*
dsolve({*de*, *cond*}, *y*(*x*), *series*); *convert*(%, *polynom*); *y2* := *rhs*(%)

$$y(x) = 1 + x + x^2 + \frac{1}{6} x^3 + O(x^4)$$

$$y(x) = 1 + x + x^2 + \frac{1}{6} x^3$$

$$y2 := 1 + x + x^2 + \frac{1}{6} x^3$$

> *# Построение графиков найденных решенийplot*([*y1*, *y2*], *x* =−3 ..3)



На рисунке видно, что наилучшее приближение точного решения степенным рядом достигается примерно на интервале $-1<x<1$.

Численное решение дифференциальных уравнений.

Для того чтобы найти численное решение дифференциального уравнения (задачи Коши или краевой задачи) в команде *dsolve* следует указать параметр *type=numeric* (или просто *numeric*). Тогда команда решения дифференциального уравнения будет иметь вид *dsolve*(*de*, *var*, *type*=*numeric*, *options*), где *de* – уравнение, *var* – неизвестная функция, *options* – параметры, позволяющие указать метод численного интегрирования дифференциального уравнения. В Maple реализованы, например, такие методы: *method=rkf45* – метод Рунге-Кутты-Фельберга 4-5-ого порядка (установлен по умолчанию); *method=dverk78* – метод Рунге-Кутта 7-8 порядка; *method=classical* – классический метод Рунге-Кутты 3-го порядка; *method=gear* и *method=mgear* – одношаговый и многошаговый методы Гира.

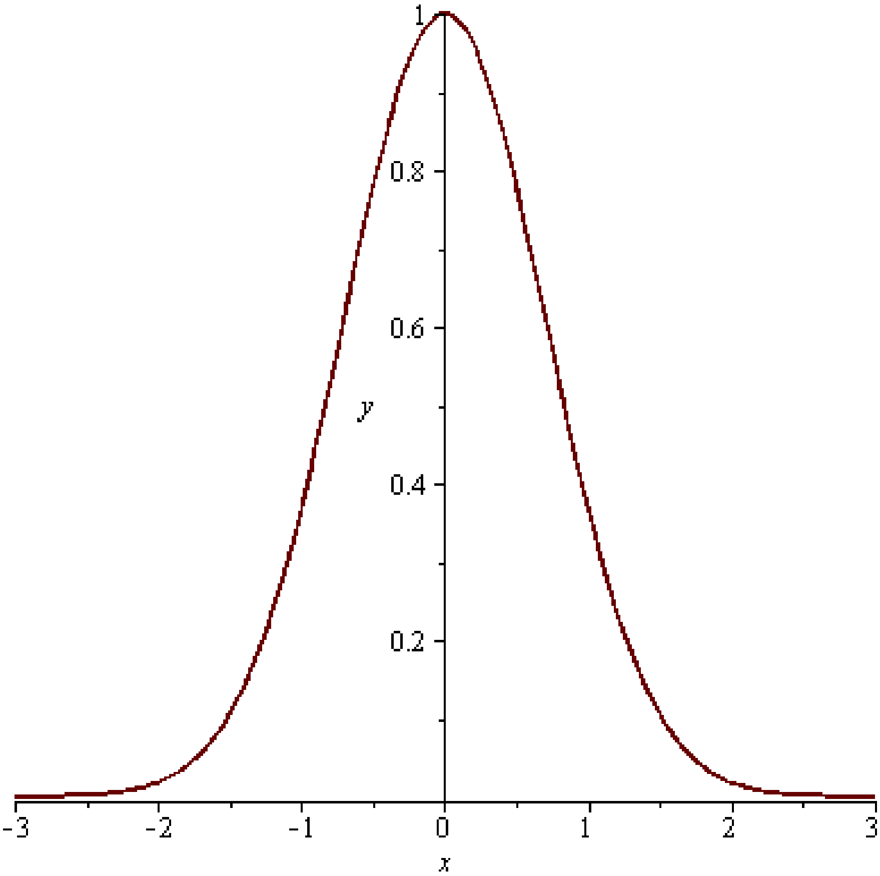
График численного решения дифференциального уравнения можно построить с помощью команды *odeplot*(*dsn*, [*x*,*y*(*x*)], *x*=*x*1..*x*2), где в качестве выражения используется результат команды *dsn*:=*dsolve*({*de*,*cond*}, *y*(*x*), *numeric*) численного решения, после нее в квадратных скобках указывают переменную и неизвестную функцию [x,y(x)], а также интервал x=x1..x2 для построения графика. Заметим, что требуется подключения пакета *plots* для ее выполнения.

Найдем численное решение уравнения $y' = -2xy$ при условии $y(0) = 1$ и сравним с точным, построив графики:

```
> with(plots) : dsolve({diff(y(x), x) = -2·x·y(x), y(0) = 1});# точное частное решение
```

$$y(x) = e^{-x^2}$$

```
> # Задание графика полученной сеточной функции
  p1 := odeplot(dsolve({diff(y(x), x) = -2·x·y(x), y(0) = 1}, numeric), [x, y(x)], x = -3 ..3) :
> # Задание графика точного решенияp2 := plot(e^(-x^2), x = -3 ..3) :
> display(p1, p2)# построение обоих графиков в одной системе координат
```



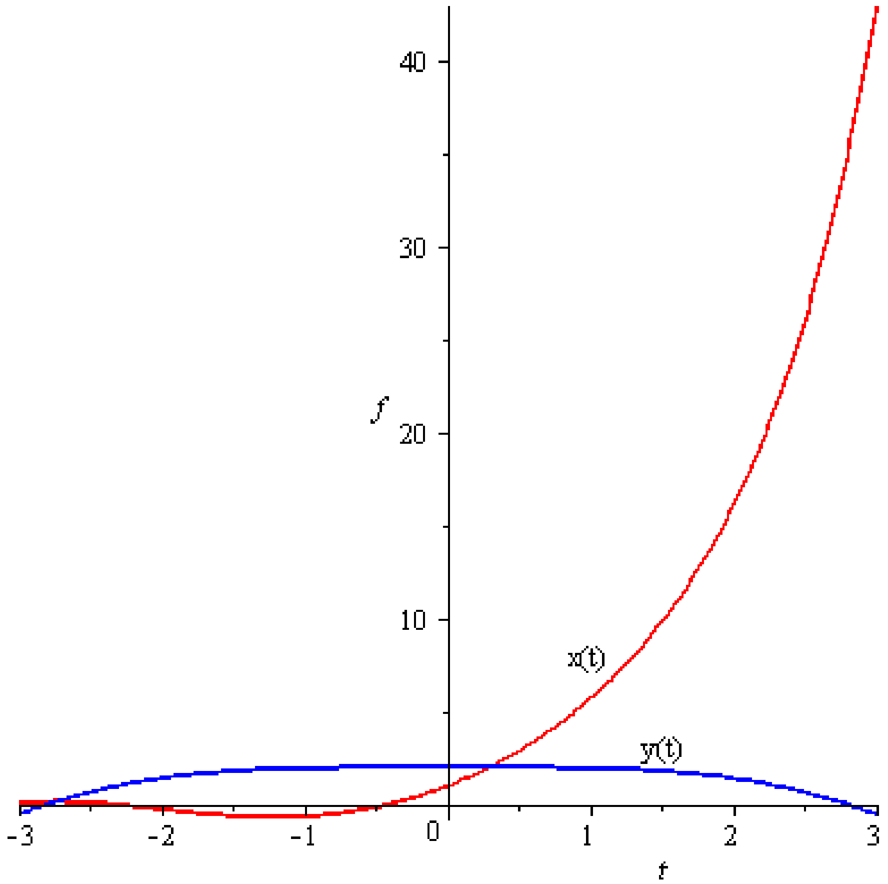
Видим, что найденные решения практически совпадают.

Формат команды *dsolve* для численного решения системы ОДУ отличается незначительно: *dsolve*({*sde*, *cond*},{*vars*}, *type*=*numeric*, *options*), где *sde* – множество уравнений системы, *cond* – начальные условия, *vars* – множество неизвестных функций, *options* – параметры, позволяющие указать метод численного интегрирования дифференциальных уравнений.

Найдем решение системы $x'(t) = y(t)\cos(t) + x(t),\ y'(t) = \sin(t) - t$ ОДУ с начальными условиями $x(0) = 1,\ y(0) = 2$ с помощью численного метода и построим графики найденных функций в одной системе координат:

```
> dsn := diff(x(t), t) = y(t)·cos(t) + x(t), diff(y(t), t) = sin(t) − t :# ввод системы
  cond := x(0) = 1, y(0) = 2 :# ввод начальных условий
  SF := dsolve({dsn, cond}, {x(t), y(t)}, numeric) :

> with(plots) :# подключение графического пакета
  p1 := odeplot(SF, [t, x(t)], t=−3 ..3, color = red) :
  p2 := odeplot(SF, [t, y(t)], t=−3 ..3, color = blue) :
  p3 := textplot([[1, 8, "x(t)"],[1.5,3,"y(t)"]]) :
  display(p1, p2, p3);
```



Для численного решения задачи Коши, построения графиков решения и фазовых портретов в Maple имеется специальный пакет *DEtools*.

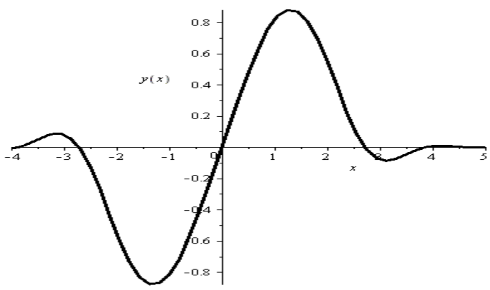
Команда *DEplot* из пакета *DEtools* строит численными методами графики решения или фазовые портреты. Эта команда аналогична команде *odeplot*, но более функциональна: она сама производит численное решение дифференциального уравнения. Основные параметры *DEplot* похожи на параметры *odeplot*: *DEplot*(*sde*, *vars*, *range*, x=$x1..x2$,…, y=$y1..y2$, *cond*, *options*), где *sde* – система дифференциальных уравнений (или одно уравнение); *vars* – множество неизвестных функций (или одна функция); *range* – диапазон изменения независимой переменной; *cond* – начальные условия; $x=x1..x2$ и $y=y1..y2$ – диапазоны изменения функций; *options* – дополнительные (необязательные) параметры.

Наиболее востребованные параметры: *linecolor* − цвет линии; *scene*=[*x*,*y*] – определение зависимости для вывода графика; *iterations*=*n* − число итераций, необходимое для повышения точности вычислений (по умолчанию *n*=1); *stepsize*=*number* − число, равное расстоянию между точками на графике (по умолчанию *number*= ($x2$–$x1$)/48); *obsrange*=*true*/*false* – указатель прерывания вычислений в случае выхода графика решения за установленный диапазон (по умолчанию имеет значение *true*); различные параметры для анимации (см. справочные материалы о команде).

Построим график решения дифференциального уравнения $y'' + xy' + x^2 y = 0$ с начальными условиями $y(0) = 0, y'(0) = 1$ в интервале $[-4;5]$.

> $with(DEtools):$

$DEplot\big(diff(y(x), x\$2) + x\cdot diff(y(x), x) + x^2\cdot y(x) = 0, y(x), x = -4..5, [[y(0) = 0, \mathrm{D}(y)(0) = 1]]\big)$



>

С помощью команды *DEplot* можно построить фазовый портрет для системы двух дифференциальных уравнений $x'(t) = f_1(t, x, y), y'(t) = f_2(t, x, y)$, если в параметрах данной команды указать значение *scene*=[*x*,*y*].

Если система дифференциальных уравнений является автономной, то на фазовом портрете будет построено поле направлений в виде стрелок. Размер стрелок регулируется параметром *arrows*.

Для того чтобы нарисовать весь фазовый портрет, необходимо для каждой фазовой траектории указывать начальные условия: например, для системы двух дифференциальных уравнений первого порядка несколько начальных условий в команде *DEplot* указываются после задания диапазона изменения независимой переменной t: [[$x(t0)=x0$, $y(t0)=y0$], [$x(t0)=x1$, $y(t0)=y1$],…].

Начальные условия можно задавать в более компактной форме: [$t0$, $x0$, $y0$], где $t0$ – точка, в которой задаются начальные условия, $x0$ и $y0$ – значения искомых функций в точке $t0$.

Фазовый протрет системы двух дифференциальных уравнений первого порядка можно также построить с помощью команды *phaseportrait*(*sde*, [*x*,*y*],*x*1..*x*2,[[*cond*]]), где *sde* – система двух дифференциальных уравнений первого порядка, [*x*,*y*] – имена искомых функций, *x*1..*x*2 – интервал, на котором следует построить фазовый портрет, а в скобках указываются начальные условия.

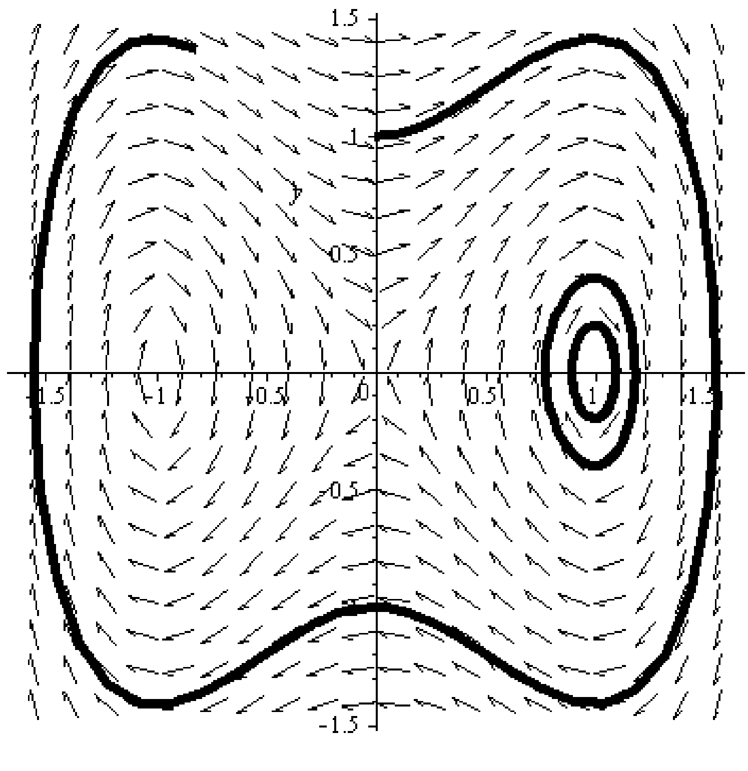
Напомним, что рассмотренные команды находятся в пакете *DEtools*, который должен быть предварительно загружен.

Построим с помощью команды *DEplot* фазовый портрет системы дифференциальных уравнений $x'=\frac{1}{2}y,\ y'=x-x^3$ для нескольких наборов начальных условий: $x(0)=1$, $y(0)=0.2$; $x(0)=0$, $y(0)=1$; $x(0)=1$, $y(0)=0.4$.

> $with(DEtools)$:

$$DEplot\left(\left\{diff(x(t),t)=\frac{1}{2}\cdot y(t), diff(y(t),t)=x(t)-(x(t))^3\right\}, [x(t),y(t)], t=0..10, [[0,1,0.2],[0,0,1],[0,1,0.4]]\right)$$

>

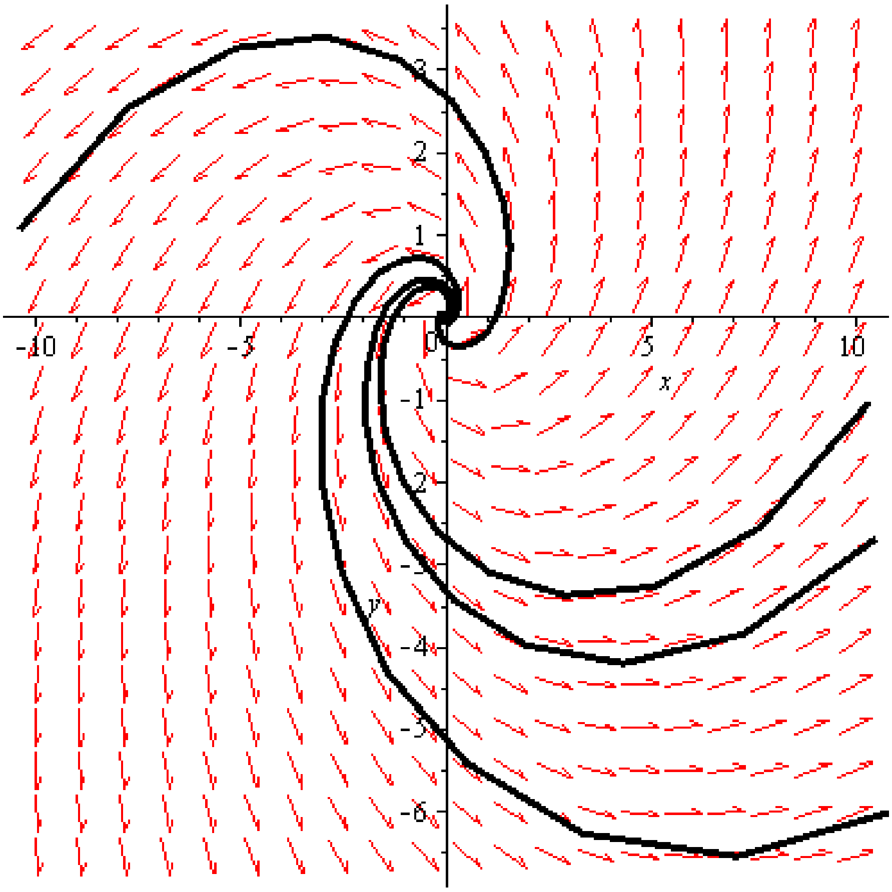


Приведем пример решения аналогичной задачи с использованием команды *phaseportrait* (определите по тексту программы систему и начальные данные):

> $restart;\ with(DEtools):$

$sde := \mathrm{D}(x)(t) = x(t) - 2\cdot y(t), \mathrm{D}(y)(t) = x(t) + y(t):$
$phaseportrait(\{sde\}, [x(t), y(t)], t = -5..5, [[0, 1, 2], [0, -3, -2], [0, 2, -4], [0, -1, -2]], x = -10..10)$

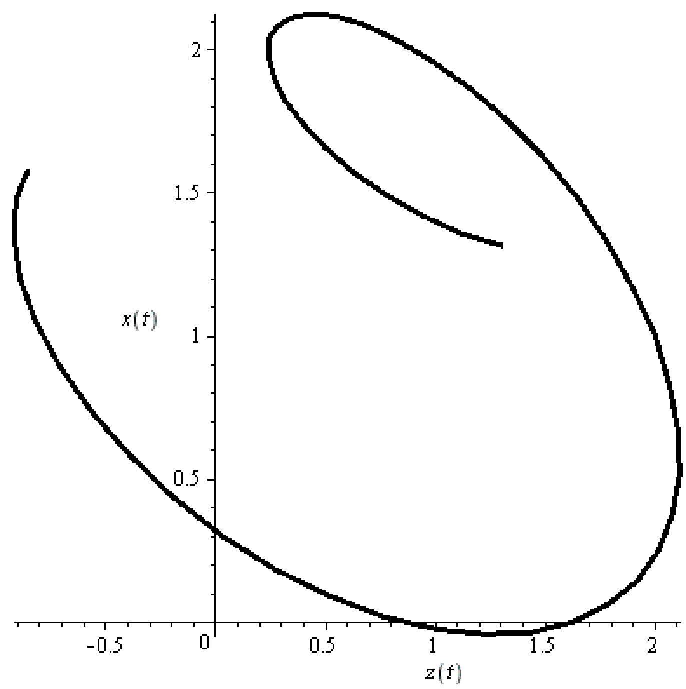
>



Заметим, что при вводе последней системы использовались дифференциальные операторы D(*f*). Предлагается проследить за изменением рисунка при задании других начальных условий, диапазона изменения переменной и размеров координатных осей.

Наконец, приведем пример системы с тремя неизвестными функциями $x(t), y(t), z(t)$, отражающий зависимость *z* от *x* (параметр *scene* команды *phaseportrait*):

> $phaseportrait([diff(x(t), t) = y(t) - z(t), diff(y(t), t) = z(t) - x(t), diff(z(t), t) = x(t) - y(t)\cdot 2], [x(t), y(t), z(t)], t = -2..2, [[x(0) = 1, y(0) = 0, z(0) = 2]], scene = [z(t), x(t)])$



x(t)
z(t)
2
1.5
1
0.5
-0.5
0
0.5
1
1.5
2