Обыкновенные дифференциальные уравнения

Известно, что с помощью дифференциальных уравнений можно описать задачи движения системы взаимодействующих материальных точек, химической кинетики, электрических цепей, и др. Конкретная прикладная задача может приводить к дифференциальному уравнению любого порядка, или к системе уравнений любого порядка. При этом известно, что произвольное дифференциальное уравнение p-го порядка

$$y^{(p)} = f(x, y, y', y'', \dots y^{(p-1)}) \qquad (1)$$

при помощи замены $y^{(k)} \equiv y_k(x)$ можно свести к системе p уравнений 1-го порядка:

$$\begin{cases} y'_k(x) = y_{k+1}(x), \\ y'_{p-1} = f(x, y_0, y_1, \dots, y_{p-1}) \end{cases} \quad 0 \le k \le p-2 \qquad (2)$$

где $y_0(x) \equiv y(x)$. Используя алгоритм перехода от (1) к (2), можно любую систему любого порядка свести к системе уравнений 1-го порядка.

Покажем такое преобразование на примере уравнения Бесселя: $y'' + \frac{y'}{x} + \left(1 - \frac{p^2}{x^2}\right) y = 0$.

Предполагая тождественную замену $y_1(x) \equiv y(x)$ представим систему дифференциальных уравнений в следующем виде:

$$\begin{cases} y'_1(x) = y_2(x) \\ y'_2(x) = \left(\frac{p^2}{x^2} - 1\right) y_1(x) - \frac{y_2(x)}{x} \end{cases}$$

тогда $y_1''(x) = y_2'(x)$

Аналогично произвольную систему дифференциальных уравнений любого порядка можно заменить некоторой эквивалентной системой уравнений первого порядка. Следовательно, алгоритмы численного решения достаточно реализовать для решения системы дифференциальных уравнений первого порядка.

Поэтому в дальнейшем будем в основном работать с системой уравнений 1-го порядка следующего вида: $y_k'(x) = f_k(x, y_1, y_2, \dots, y_p) \quad k = 1,2,\dots,p$, (3)

которую будем записывать в сокращённой векторной форме:

$$y' = f(x, y), \qquad y = (y_1, y_2, \dots, y_n), \qquad f = (f_1, f_2, \dots, f_n) \qquad (3')$$

Известно, что система уравнений (3) и (3´) имеет множество решений, которые в общем случае зависят от p параметров с = $(c_1, \dots, c_p)$, что можно записать в виде: $y = y(x, c)$. Для выделения единственного решения необходимо наложить p дополнительных условий.

Различают три типа задач для систем дифференциальных уравнений:

- задачи Коши;
- краевые задачи;
- задачи на собственные значения.

Задача Коши (или задача с начальными данными) предполагает дополнительные условия вида: $y_k(x_0) = y_{0,k}, \quad k = 1,\dots,p$ (4)

т.е. в точке x_0 задаются значения всех функций системы уравнений (3). Условие (4) можно истолковать как задание начальной точки $(x_0, y_{0,1}, \dots, y_{0,p})$ искомой интегральной кривой в $p+1$-мерном пространстве $(x_0, y_1, \dots, y_p)$. Решение системы уравнений (3) обычно требуется найти на отрезке $[x_0, X]$.

Краевая задача - это задача отыскания частного решения системы ОДУ на отрезке $a \le x \le b$, в которой дополнительные условия налагаются на значения функции $y_k(x)$ более чем в одной точке этого отрезка.

Задача на собственные значения. Кроме искомых функций и их производных в уравнение входят дополнительно m неизвестных параметров $\lambda_1, \lambda_2, \ldots \lambda_m$, которые являются собственными значениями. Для единственности решения на некотором интервале необходимо задать $p+m$ граничных условий.

Методы решения дифференциальных уравнений можно классифицировать на ***точные, приближенные и численные.***

Точные методы позволяют выразить решения дифференциальных уравнений либо через элементарные функции, либо с помощью квадратур от элементарных функций. Точные методы решения уравнений (3), (4) изучаются в курсе обыкновенных дифференциальных уравнений.

Приближенными будем называть методы, в которых решение $y(x)$ получается как предел некоторой последовательности $y_n(x)$, при этом $y_n(x)$ выражаются либо в аналитическом виде с помощью элементарных функций, либо с помощью квадратур этих элементарных функций.

Численные методы представляют собой алгоритмы вычисления приближенных значений искомой функции $y(x)$ в узлах некоторой сетки значений аргумента $x_1, \ldots x_n$. Основным недостатком численных методов является то, что они позволяют найти частное решение, например решение задачи Коши, но не общее решение $y = y(x, c)$ уравнения (3). Этот недостаток компенсируется тем, что численные методы универсальны и могут быть использованы для решения широкого класса дифференциальных уравнений.

В большинстве случаев необходимость численного решения систем дифференциальных уравнений возникает в случае, когда *аналитическое решение* найти либо невозможно, либо нерационально, а *приближенное решение* (в виде набора интерполирующих функций) не дает требуемой точности.

Численные методы могут быть использованы только для задач, решение которых существует и единственно, т.е. если задача ***корректно*** поставлена. Однако условие корректности необходимо, но не достаточно для численного решения задачи. Необходимо ещё, чтобы задача была ***хорошо обусловлена***, т.е. малые шевеления в начальных (входных) данных приводили к малым изменениям в решениях. Если это условие не выполнено, то считается, что задача ***плохо обусловлена (слабо устойчива),*** т.е. небольшие погрешности входных данных или метода могут сильно исказить решение.

Приведём простой **пример плохо обусловленной задачи**. Необходимо численно найти решение следующей задачи:

$$y' = y \qquad 0 \le x \le 100 \qquad y(0) = y_0 = 0 \tag{5}$$

Задача (5) имеет точное общее решение: $y = y_0 e^x$ и частное $y \equiv 0$, входным значение для задачи является параметр y_0 . Плохая обусловленность задачи (5) выражается в том, что если пошевелить входной параметр, сделать его равным не нулю, а 10^{-6}, то решение в точке 100 сильно изменится, т.е. $y(100) = 10^{-6} \cdot 2{,}718^{100} = 2{,}688 \cdot 10^{37}$.

Численные методы решения дифференциальных уравнений первого порядка

Общий вид дифференциального уравнения $F(x, y, y') = 0.$ (6)

Нормальная форма дифференциального уравнения $y' = f(x, y),$ (7)

где y=y(x) - неизвестная функция, подлежащая определению,

f(x,y) - правая часть дифференциального уравнения в нормальной форме, равная первой производной функции y(x). В функцию f(x,y) помимо аргумента x входит и сама неизвестная функция y(x).

Пример:

$x \cdot y' - (x^2 - 1) \cdot y = 0$ - общий вид дифференциального уравнения первого порядка,

$y' = \frac{x^2-1}{x} \cdot y$ - нормальная форма этого же уравнения.

Если неизвестная функция y зависит от одного аргумента x, то дифференциальное уравнение вида $y' = f(x, y),$ называется **обыкновенным** дифференциальным уравнением. Если функция y зависит от нескольких аргументов, то такое дифференциальное уравнение называется дифференциальным уравнением в **частных производных.** Общим решением обыкновенного дифференциального уравнения $y' = f(x, y),$ является семейство функций y=y(x,c) (рис. 1):

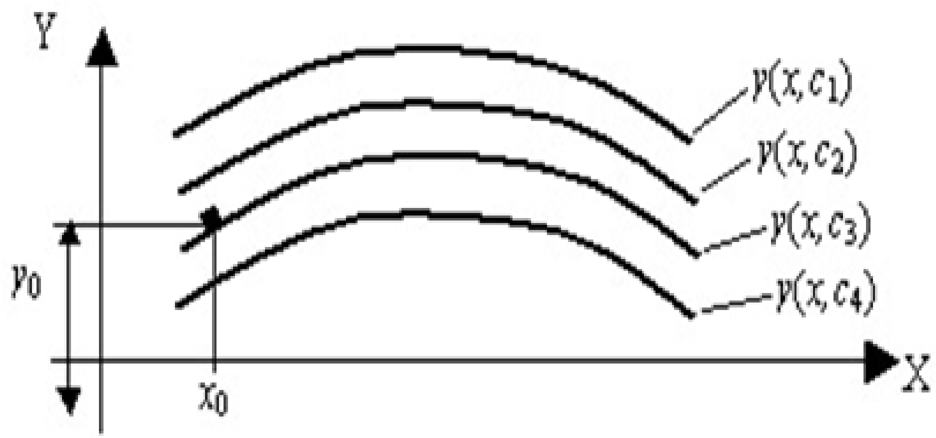


Рис. 1. Графическая иллюстрация решения ОДУ

При решении прикладных задач ищут частные решения дифференциальных уравнений. Выделение частного решения из семейства общих решений осуществляется с помощью задания начальных условий: $y|_{x=x_0} = y_0$, (8)
т.е. начальной точки с координатами (x_0, y_0).

Нахождение частного решения дифференциального уравнения $y' = f(x,y)$, удовлетворяющего начальному условию $y|_{x=x_0} = y_0$, называется задачей Коши.

В численных методах задача Коши ставится следующим образом: найти табличную функцию $y_i = f(x_i), i = \overline{1,n}$, которая удовлетворяет заданному дифференциальному уравнению (7) и начальному условию (8) на отрезке [a,b] с шагом h, то есть найти таблицу

i	x	y
0	x_0	y_0
1	x_1	y_1
2	x_2	y_2
3	x_3	y_3
...	...	...
n	x_n	y_n

Здесь

h - шаг интегрирования дифференциального уравнения,

$a=x_0$ - начало участка интегрирования уравнения,

$b=x_n$ - конец участка,

$n=(b-a)/h$ - число шагов интегрирования уравнения.

На графике (рис. 2) решение задачи Коши численными методами представляется в виде совокупности узловых точек с координатами (x_i, y_i), $i=\overline{1,n}$.

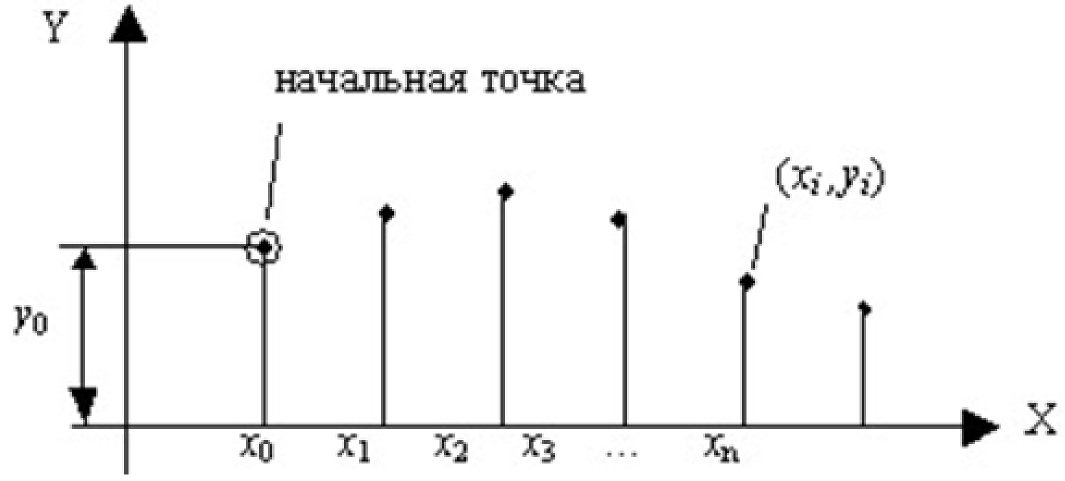


Рис. 2. Графическая иллюстрация решения задачи Коши

Теорема существования и единственности решения задачи Коши

Пусть функция $f(x,y)$ определена и непрерывна на множестве точек $\{x_0 \le x \le X, -\infty < y < \infty\}$. Предположим также, что она удовлетворяет **условию Липшица**: $|f(x,y_1) - f(x,y_2)| \le L|y_1 - y_2|$ для всех $x_0 \le x \le X$ и произвольных y_1, y_2, где L - некоторая константа (постоянная Липшица). Тогда для каждого начального значения y_0 существует единственное решение $y(x)$ задачи Коши, определенное на отрезке $[x_0, X]$.

Геометрически задача интегрирования дифференциальных уравнений состоит в нахождении интегральных кривых, которые в каждой своей точке имеют заданное направление касательной. Совокупность решений образует *поле направлений*. Заданием начального условия мы выделяем из семейства решений ту единственную кривую, которая проходит через фиксированную точку (x_0, y_0).

Методы Рунге - Кутта

Наиболее эффективными и часто встречаемыми методами решениями задачи Коши являются методы Рунге - Кутта. Они основаны на аппроксимации искомой функции y(x) в пределах каждого шага многочленом, который получен при помощи разложения функции y(x) в окрестности шага h каждой i-ой точки в ряд Тейлора:

$$y(x_i+h)=y(x_i)+h\cdot y'(x_i)+\frac{h^2}{2!}y''(x_i)+ \\ +\frac{h^3}{3!}y'''(x_i)+\frac{h^4}{4!}y^{(4)}(x_i)+\frac{h^5}{5!}y^{(5)}(x_i)+\ldots \qquad (9)$$

Усекая ряд Тейлора в различных точках и отбрасывая правые члены ряда, Рунге и Кутт получали различные методы для определения значений функции y(x) в каждой узловой точке. Точность каждого метода определяется отброшенными членами ряда.

Метод Рунге - Кутта 1-го порядка (метод Эйлера, метод ломаных)

Отбросим в ур.(9) члены ряда, содержащие h^2, h^3, h^4: .

Тогда $y(x_i+h)=y(x_i)+h\cdot y'(x_i)$. Учитывая, что $y'(x_i)=f(x_i,y_i)$,

получим формулу Эйлера: $$y_{i+1}=y_i+h\cdot f(x_i,y_i) \qquad (10)$$

Численный метод называется **явным**, если вычисление решения в следующей точке y_{i+1} осуществляется по явной формуле. Метод называется **одношаговым**, если вычисление решения в следующей точке y_{i+1} производится с использованием только одного предыдущего значения y_i. Метод Эйлера является явным одношаговым методом.

Так как точность методов Рунге-Кутта определяется отброшенными членами ряда (9), то точность метода Эйлера на каждом шаге составляет $\approx h^2$.

Алгоритм метода Эйлера можно построить в виде двух программных модулей: основной программы и подпрограммы ELER, реализующей метод. Здесь

(x,y)-при вводе начальная точка, далее текущие значения табличной функции,

h-шаг интегрирования дифференциального уравнения,

b-конец интервала интегрирования.

Основная программа

Подпрограмма ELER

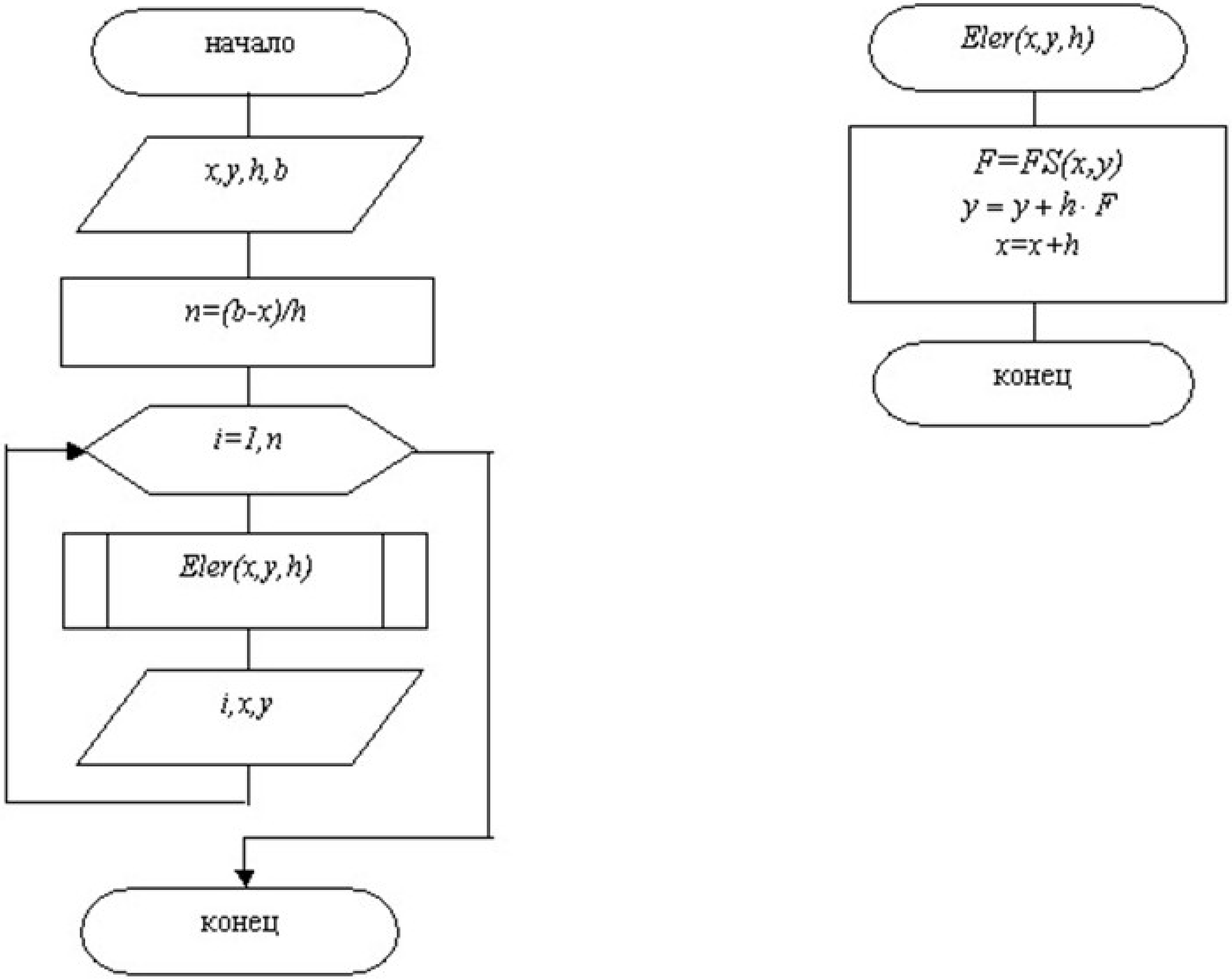


Рис. 3. Блок-схема алгоритма метода Эйлера

Рассмотрим геометрический смысл метода Эйлера. Его формула имеет вид:

$y_{i+1} = y_i + h \cdot f(x_i, y_i)$, где $f(x_i, y_i) = y'(x_i) = \operatorname{tg} \alpha_i$.

Тогда формула Эйлера принимает вид: $y_{i+1} = y_i + h \cdot \operatorname{tg} \alpha_i$, где $\operatorname{tg} \alpha_i$ - тангенс угла наклона касательной к искомой функции y(x) в начальной точке каждого шага.

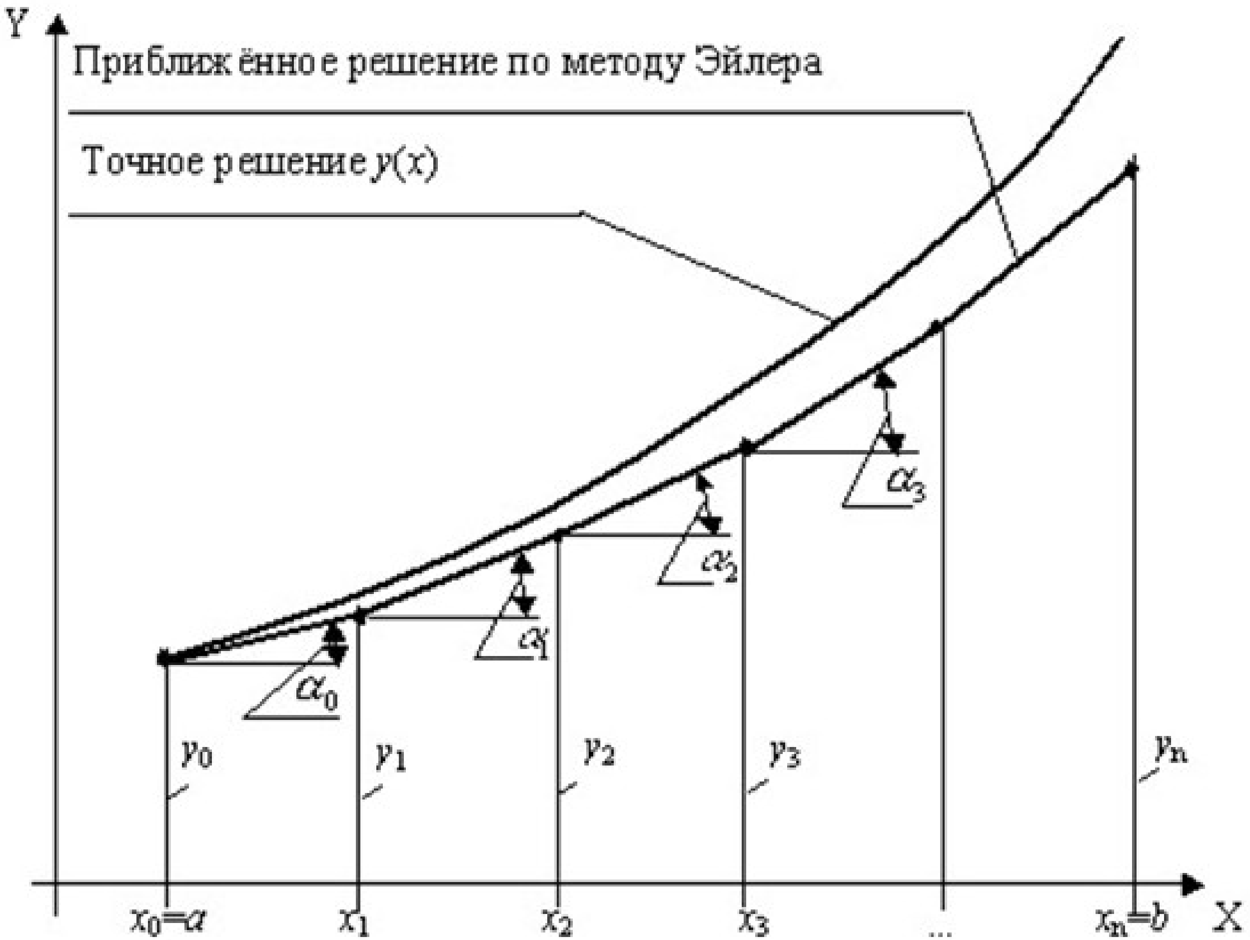


Рис. 4. Геометрический смысл метода Эйлера

В результате в методе Эйлера на графике (рис 4) вся искомая функция y(x) на участке [a,b] аппроксимируется ломаной линией, каждый отрезок которой на шаге h линейно аппроксимирует искомую функцию. Поэтому метод Эйлера получил еще название **метода ломаных.**

В методе Эйлера наклон касательной в пределах каждого шага считается постоянным и равным значению производной в начальной точке шага x_i. В действительности производная, а, значит, и тангенс угла наклона касательной к кривой y(x) в пределах каждого шага меняется. Поэтому в точке x_i+h наклон касательной не должен быть равен наклону в точке x_i. Следовательно, на каждом шаге вносится погрешность.

Первый отрезок ломаной действительно касается искомой интегральной кривой y(x) в точке (x_0,y_0). На последовательных же шагах касательные проводятся из точек (x_i,y_i), подсчитанных с погрешностью. В результате с каждым шагом ошибки накапливаются. Поэтому говорят о локальной и глобальной погрешностях метода.

Локальной погрешностью метода называется величина $l_i = y(x_{i+1}) - y_{i+1}$. Найдем величину локальной погрешности метода Эйлера:

$$l_i = y(x_{i+1}) - y_{i+1} = y(x_i) + hy'(x_i) + O(h^2) - y_i - hf(x_i, y_i) = O(h^2),$$

при условии, что $y(x_i) = y_i$. Другими словами погрешность l_i – это погрешность, которую допускает за один шаг метод, стартующий с точного решения.

Глобальной погрешностью (или просто погрешностью) численного метода называют сеточную функцию ε_h со значениями $\varepsilon_i = y(x_i) - y_i$ в узлах. В качестве меры абсолютной погрешности метода примем величину $E(h) = \max\limits_{0 \le i \le N} | y(x_i) - y_i |$.

Можно показать, что для явных одношаговых методов из того, что локальная погрешность имеет вид $l_i = Ch^{p+1}$, следует, что $E(h) = Mh^p$, где C и M - некоторые константы. Таким образом, метод Эйлера является методом **первого порядка точности**. Для нахождения решения задачи Коши с заданной точностью ε требуется найти такое приближенное решение y^h, для которого величина глобальной погрешности $E(h) \le \varepsilon$. Так как точное решение задачи неизвестно, погрешность оценивают с помощью правила Рунге.

Правило Рунге оценки погрешностей. Для практической оценки погрешности проводят вычисления с шагами h и h/2. За оценку погрешности решения, полученного с шагом h/2, принимают величину, равную $\max\limits_i \left| \dfrac{y_i^h - y_{2i}^{h/2}}{2^p - 1} \right|$, где p - порядок метода.

Основной недостаток метода Эйлера - систематическое накопление ошибок. Поэтому метод Эйлера рекомендуется применять для решения дифференциальных уравнений при малых значениях шага интегрирования h.

Метод Рунге - Кутта 2-го порядка (метод Эйлера-Коши)

Метод Эйлера обладает медленной сходимостью, поэтому чаще применяют методы более высокого порядка точности. Отбросим в ур.(9) члены ряда, содержащие h_3, h_4, h_5:.

Тогда

$$y(x_i + h) = y(x_i) + h \cdot y'(x_i) + \frac{h^2}{2!} \cdot y''(x_i). \quad (11)$$

Чтобы сохранить член ряда, содержащий h^2, надо определить вторую производную $y''(x_i)$. Ее можно аппроксимировать разделенной разностью 2-го порядка:

$$y''(x_i) = \frac{\Delta y'}{\Delta x} = \frac{y'(x_i + h) - y'(x_i)}{h}$$

Подставляя это выражение в (11), получим:

$$y(x_i + h) = y(x_i) + h \cdot y'(x_i) + \frac{h}{2} \cdot \frac{y'(x_i+h) - y'(x_i)}{h} =$$
$$= y(x_i) + \frac{h}{2} y'(x_i) + \frac{h}{2} y'(x_i + h).$$

Окончательно, модифицированная или уточненная формула Эйлера имеет вид:

$$y_{i+1} = y_i + \frac{h}{2} f(x_i, y_i) + \frac{h}{2} f(x_{i+1}, y_{i+1}). \quad (12)$$

Как видно, для определения функции y(x) в точке i+1 необходимо знать значение правой части дифференциального уравнения $f(x_{i+1}, y_{i+1})$ в этой точке, для определения которой необходимо знать предварительное значение y_{i+1}.

Для определения предварительного значения y_{i+1} воспользуемся формулой Эйлера. Тогда все вычисления на каждом шаге по модифицированной или уточненной формуле Эйлера будем выполнять в два этапа:

На первом этапе вычисляем предварительное значение $y_{i+1}^{Э}$ по формуле Эйлера

$$y_{i+1}^{Э} = y_i + hf(x_i, y_i).$$

На втором этапе уточняем значение y_{i+1} по модифицированной или уточненной формуле Эйлера $$y_{i+1} = y_i + \frac{h}{2}f(x_i, y_i) + \frac{h}{2}f(x_{i+1}, y_{i+1}^{Э}).$$

Точность метода определяется отброшенными членами ряда Тейлора (9), т.е. точность уточненного или модифицированного метода Эйлера на каждом шаге $\approx h^3$.

Рассмотрим геометрический смысл модифицированного метода Эйлера.

Так как
$$f(x_i, y_i) = y'(x) = \operatorname{tg}\alpha_i,$$
$$f(x_{i+1}, y_{i+1}) = y'(x_{i+1}) = \operatorname{tg}\alpha_{i+1},$$

то модифицированную формулу Эйлера можно представить в виде:

$$y_{i+1} = y_I + \frac{h}{2}\operatorname{tg}\alpha_i + \frac{h}{2}\operatorname{tg}\alpha_{i+1},$$

где $\operatorname{tg}\alpha_i$ - тангенс угла наклона касательной к искомой функции y(x) в начальной точке каждого шага, $\operatorname{tg}\alpha_{i+1}$ - тангенс угла наклона касательной к искомой функции y(x) в конечной точке каждого шага.

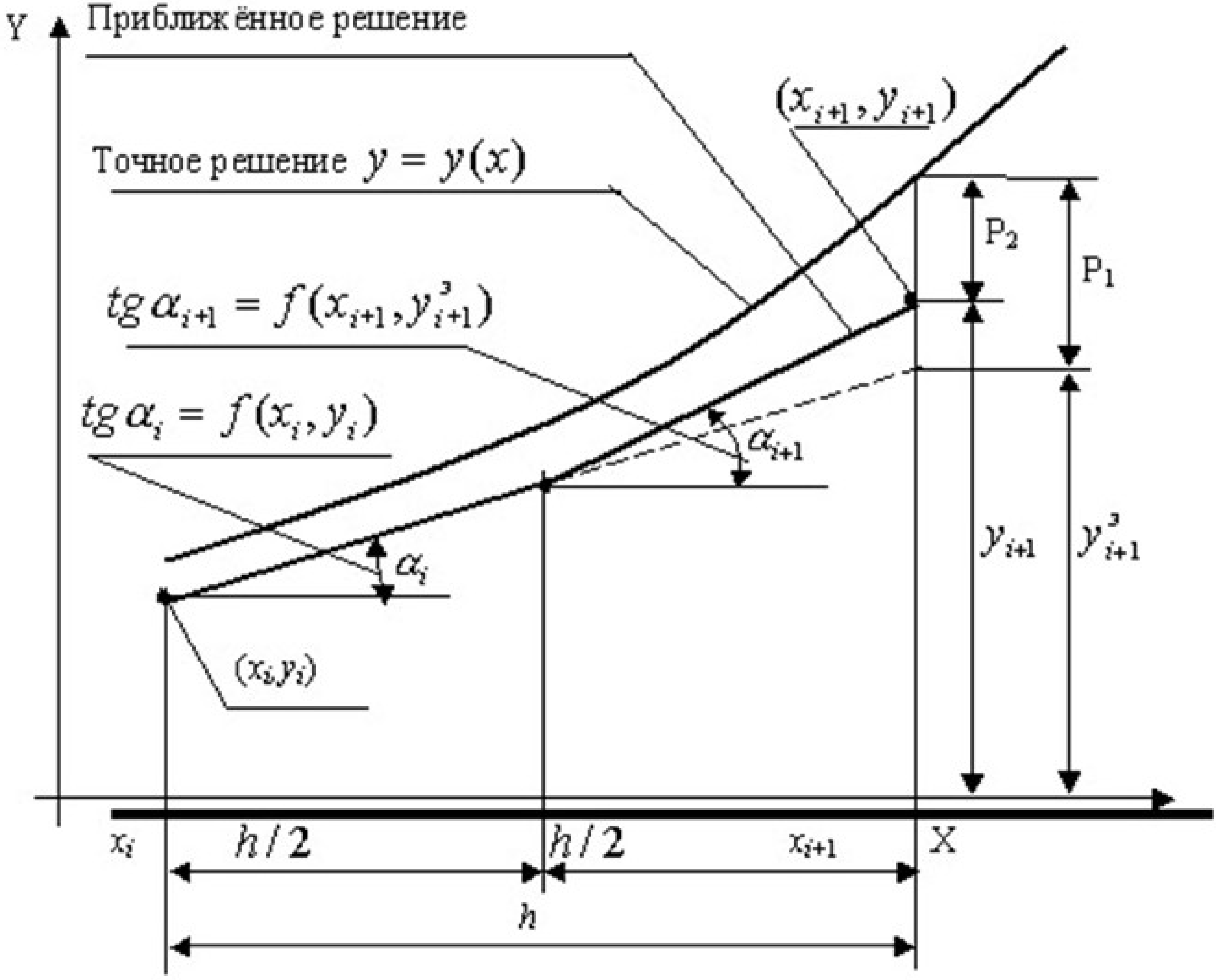


Рис. 5. Геометрический смысл модифицированного метода Эйлера

Здесь P_1 - накопленная ошибка в $(i+1)^{й}$ точке по методу Эйлера,

P_2 - накопленная ошибка в $(i+1)^{й}$ точке по модифицированному методу Эйлера.

Как видно из рис.5, в первой половине каждого шага, то есть на участке $[x_i, x_i+h/2]$, искомая функция y(x) аппроксимируется прямой, которая выходит из точки (x_i, y_i) под углом, тангенс которого $\operatorname{tg} \alpha_i = f(x_i, y_i)$.

Во второй половине этого же шага, т.е. на участке $[x_i + h/2, x_i + h]$, искомая функция y(x) аппроксимируется прямой, которая выходит из точки с координатами

$$x = x_i + h/2,$$
$$y = y_i = \frac{h}{2} \cdot f(x_i, y_i)$$

под углом, тангенс которого $\operatorname{tg} \alpha_{i+1} = f(x_{i+1}, y^{Э}_{i+1})$.

В результате в модифицированном методе Эйлера функция y(x) на каждом шаге аппроксимируется не одной прямой, а двумя.

Алгоритм модифицированного метода Эйлера можно построить в виде двух программных модулей: основной программы и подпрограммы MELER, реализующей метод (рис. 6).

Основная программа

Подпрограмма MELER

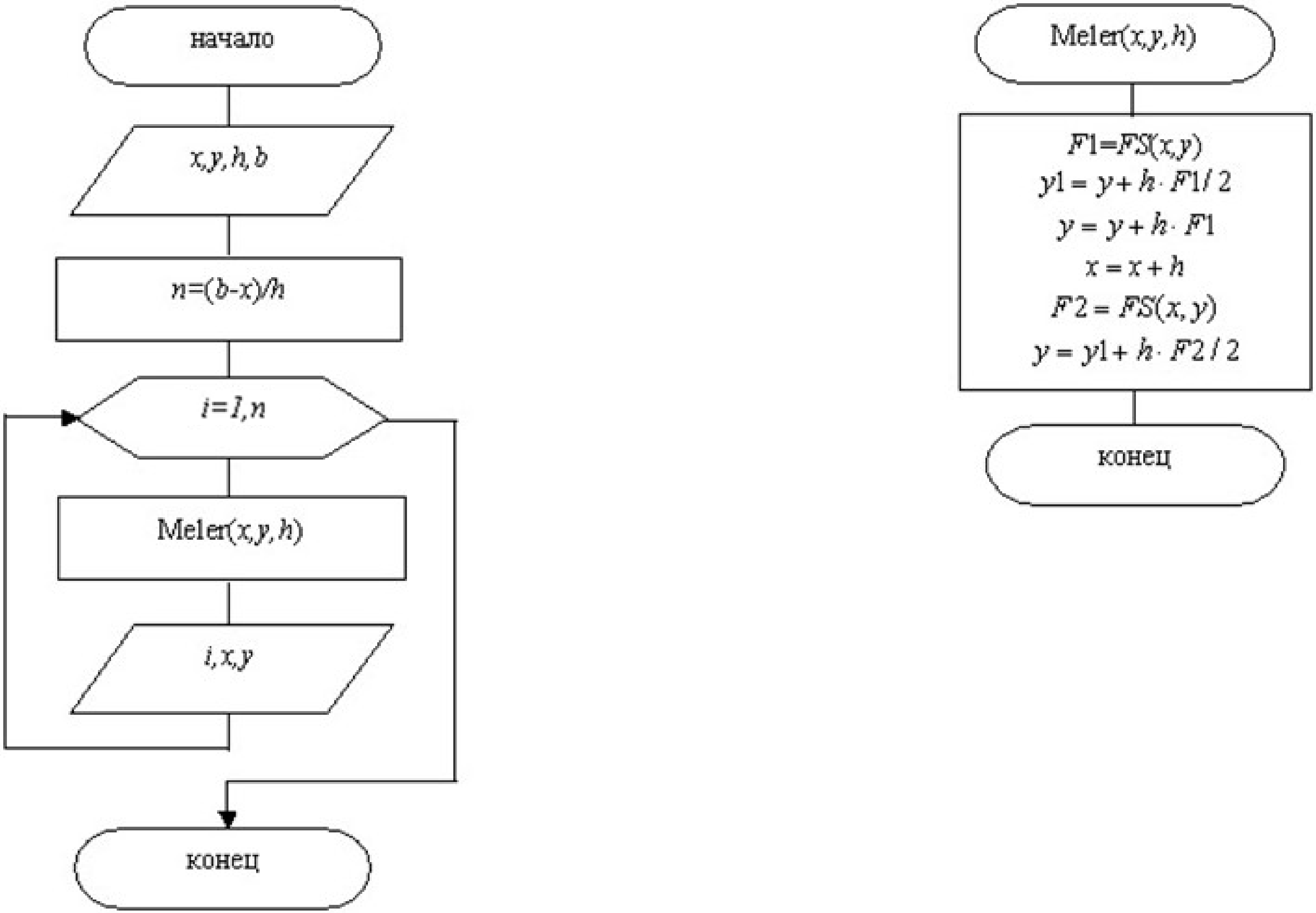


Рис. 6. Схема алгоритма модифицированного метода Эйлера

Здесь (x,y)-при вводе начальная точка, далее текущие значения табличной функции,

h-шаг интегрирования дифференциального уравнения,

b-конец интервала интегрирования.

Усовершенствованный метод Эйлера-Коши с уточнением

Данный метод базируется на предыдущем, однако здесь апостериорная погрешность контролируется на каждом шаге вычисления. Как и в предыдущем случае, рассматриваем задачу Коши на сетке с постоянным шагом *h*. Грубое значение $y_{k+1}^{\{0\}}$ вычисляется по формуле Эйлера:

$$y_{k+1} = y_k + h \cdot f(x_k, y_k)$$

и затем итерационным циклом уточняется по формуле:

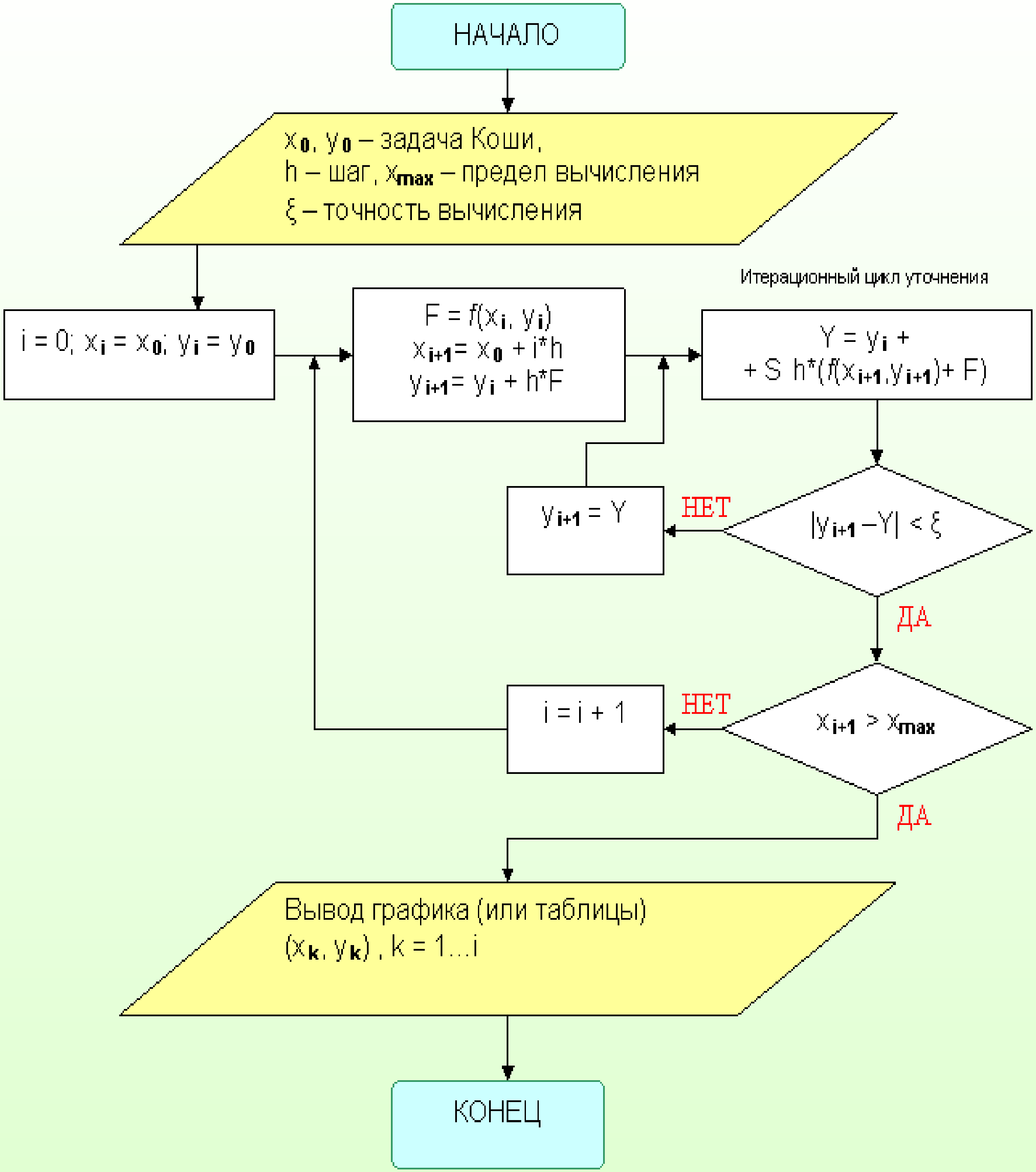
$$y_{k+1}^{\{m+1\}} = y_k^{\{m\}} + \frac{1}{2} h \cdot \left(f\left(x_k, y_k^{\{m\}}\right) + f\left(x_{k+1}, y_{k+1}^{\{m\}}\right) \right)$$

, где m – номер итерации.

Итерационный цикл повторяется до тех пор, пока $\left| y_{k+1}^{\{m+1\}} - y_{k+1}^{\{m\}} \right| < \varepsilon$.

Данная формула также легко обобщается на решение систем ОДУ. Априорная погрешность метода при $m = 1$ на каждом шаге порядка h^3.

Рис.6’. Блок-схема метода Эйлера-Коши с уточнением



Метод Рунге - Кутта 4-го порядка

Самое большое распространение из всех численных методов решения дифференциальных уравнений получил метод Рунге-Кутта 4-го порядка. В литературе он известен как просто метод Рунге-Кутта.

В этом методе на каждом шаге интегрирования дифференциальных уравнений искомая функция y(x) аппроксимируется рядом Тейлора (9), содержащим члены ряда с h^4:

$$y(x_i + h) = y(x_i) + h \cdot y'(x_i) + \frac{h^2}{2!} y''(x_i) + \frac{h^3}{3!} y^{(3)}(x_i) + \frac{h^4}{4!} y^{(4)}(x_i) \ldots$$

В результате ошибка на каждом шаге имеет порядок h^5.

Для сохранения членов ряда, содержащих h_2,h_3,h_4 необходимо определить вторую y'', третью y''' и четвертую $y^{(4)}$ производные функции y(x). Эти производные аппроксимируем разделенными разностями второго, третьего и четвертого порядков соответственно.

В результате для получения значения функции y_{i+1} по методу Рунге-Кутта выполняется следующая последовательность вычислительных операций:

$$T_1 = h \cdot f(x_i, y_i),$$
$$T_2 = h \cdot f(x_i + h/2, y_i + T_1/2),$$
$$T_3 = h \cdot f(x_i + h/2, y_i + T_2/2),$$
$$T_4 = h \cdot f(x_i + h/2, y_i + T_3),$$
$$y_{i+1} = y_i + (T_1 + 2 \cdot T_2 + 2 \cdot T_3 + T_4)/6.$$

Вывод формулы не приведен. Предоставляется возможность вывод формул выполнить самостоятельно.

К достоинствам метода следует отнести высокую точность вычислений. Схемы более высокого порядка точности практически не употребляются в силу своей громоздкости. Также немаловажно, что метод является *явным*, т.е. значение y_{k+1} вычисляется по ранее найденным значениям за известное заранее число действий.

Алгоритм метода Рунге-Кутта (4-го порядка) можно построить в виде двух программных модулей: основной программы и подпрограммы Rk4, реализующей метод (рис 7).

Здесь (x,y)-при вводе начальная точка, далее текущие значения табличной функции,

h-шаг интегрирования дифференциального уравнения,

b-конец интервала интегрирования.

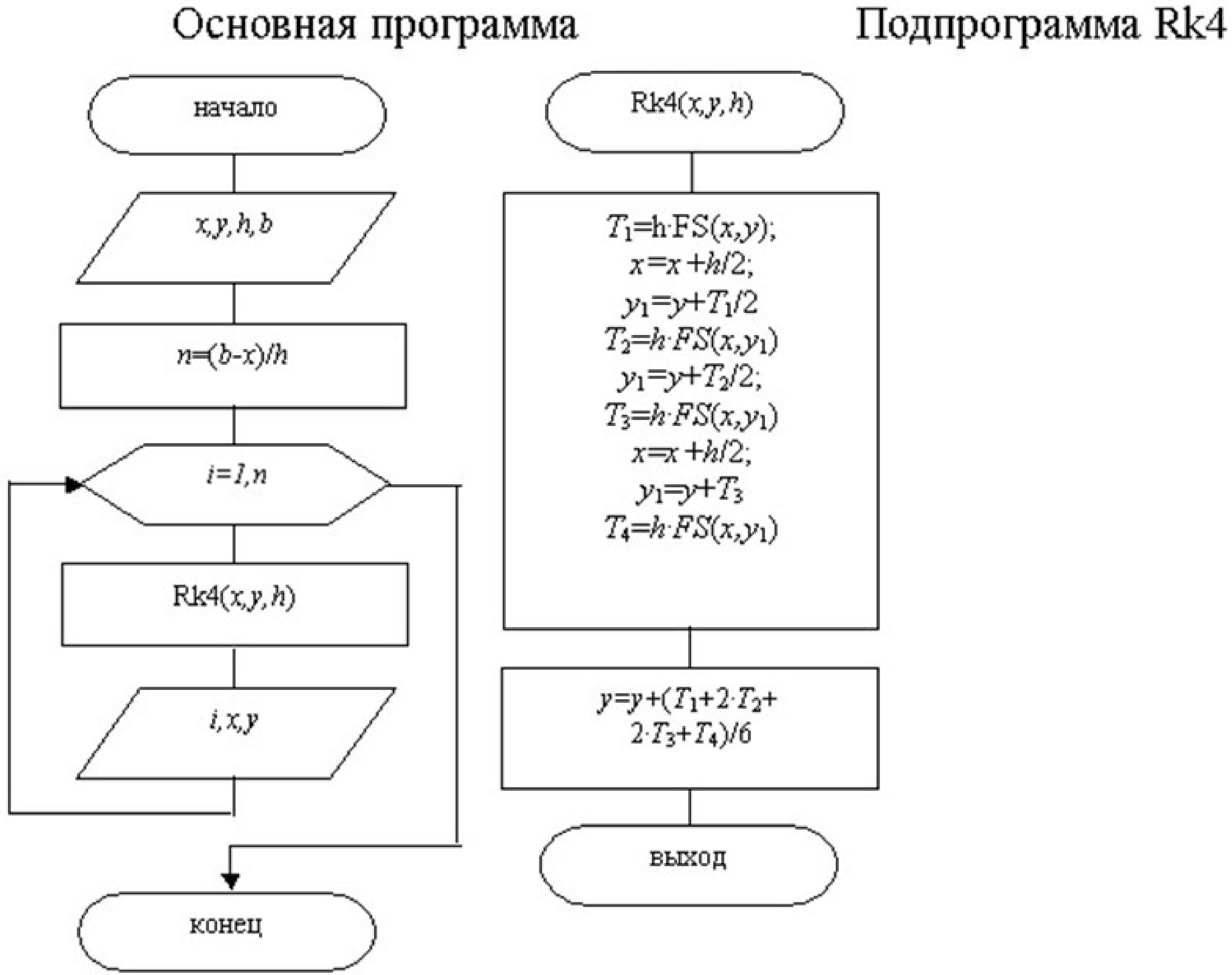


Рис. 7. Схема алгоритма метода Рунге-Кутта 4-го порядка.

Все представленные выше схемы допускают расчет с переменным шагом. Например, шаг можно уменьшить там, где функция быстро изменяется, и увеличить в обратном случае. Так, *метод Рунге-Кутты-Мерсона* позволяет оценивать погрешность на каждом шаге и, в зависимости от полученной оценки принимать решение об изменении шага. Автоматический выбор шага позволяет значительно сократить время вычислений.

Метод Рунге – Кутта - Мерсона

Этот метод отличается от метода Рунге – Кутта четвертого порядка возможностью оценивать погрешность на каждом шаге и в зависимости от этого принимать решение об изменении шага. Один из вариантов формул:

$$y(x_{n+1}) = y(x_n) + \frac{1}{2}(k_4 + k_5) + 0(h^5)$$

$$k_1 = h_3 f(x_n, y_n) \text{;} \quad h_3 = \frac{h}{3}$$

$$k_2 = h_3 f(x_n + h_3, y_n + k_1)$$

$$k_3 = hf\left[x_n + h_3, y_n + \frac{1}{2}(k_1 + k_2)\right]$$

$$k_4 = k_1 + 4h_3 f\left[x_n + \frac{h}{2}, y_n + 0.375(k_1 + k_3)\right]$$

$$k_5 = h_3 f\left[x_0 + h, y_0 + 1.5(k_4 - k_3)\right]$$

$R_{n+1} = 0.2k_4 - 0.3k_3 - 0.1k_5$ - погрешность на каждом шаге.

Пусть задана максимальна погрешность ξ. Если $|R_{n+1}| > \xi$, $h = h/2$, и (n+1) цикл расчета повторяется (с точки x_n, y_n) с новым шагом.

Если же $|R_{n+1}| < \frac{\xi}{32}$, $h = 2h$.

Автоматический выбор шага позволяет значительно сократить время решения ОДУ.

Схема метода РКМ обобщается на системы ОДУ аналогично классической схеме Рунге – Кутта.

Метод Пикара

Данный метод решает задачу Коши. Для простоты записи ограничимся одним дифференциальным уравнением. Алгоритм легко обобщается на случай системы дифференциальных уравнений путём формальной замены *y(x)* и *f(x)* на соответствующие векторы. Метод Пикара является приближенным методом, обобщающим метод последовательных приближений.

Рассмотрим задачу Коши для уравнения 1-го порядка:

$$y' = f(x, y(x)), \quad x_0 \le x \le X \quad y(x_0) = y_0 \tag{13}$$

Интегрируя дифференциальное уравнение (13) , заменим исходную задачу эквивалентной ей задачей решения интегрального уравнения Вольтерра:

$$y(x) = y_0 + \int_{x_0}^{x} f(t, y(t))dt \tag{14}$$

Решая интегральное уравнение (14) методом последовательных приближений, получим итерационный процесс Пикара:

$$y_{n+1}(x) = y_0 + \int_{x_0}^{x} f(t, y_n(t))dt \qquad n = 0,1,2,\ldots \tag{15}$$

На каждом этапе итерационного процесса Пикара в (15) интегрирование выполняется либо точно, либо численными методами.

Доказательство сходимости метода Пикара предполагает, что в некоторой ограниченной области *G(x,y)* правая часть уравнения (13) - *f(x,y)* непрерывна и удовлетворяет по переменной *y* **условию Липшица**. Т.е.

$$|f(x,\xi_1)-f(x,\xi_2)|\le L|\xi_1-\xi_2|.$$

Поскольку область *G(x,y)* ограничена, то выполняются неравенства: $|x-x_0|\le a,\quad |y-y_0|\le b$. Введём обозначения для погрешности приближенного решения: $z_n(x)=y_0(x)-y(x)$. Вычитая (14) из (15) и, используя условие Липшица, находим:

$$|z_{n+1}(x)|\le L\int_{x_0}^{x}|z_n(t)|dt \qquad (16)$$

Решая последовательно рекуррентное соотношение (16), находим:

$$|z_1|\le bL(x-x_0)\quad |z_2|\le \tfrac{1}{2}bL^2(x-x_0)^2,\ldots,\quad |z_n|\le \tfrac{1}{n!}bL^n(x-x_0)^n.$$

Из последних неравенств оценка погрешности :

$$|z_n(x)|\le \tfrac{1}{n!}b(La)^n\approx \frac{b}{\sqrt{2\pi n}}\left(\frac{eLa}{n}\right)^n \qquad (17)$$

Из оценки погрешности по ур.(17) следует, что $|z_n(x)\to 0|$ при $n\to\infty$, т.е. приближенное решение равномерно сходится к точному во всей области *G(x,y)*.

Для иллюстрации метода Пикара применим его к решению уравнения вида:

$$y'(x)=x^2+y^2\quad y(0)=0. \qquad (18)$$

Программа Data_sheet2.m рассчитывает последовательные приближённые решения ур.(18) методом Пикара. В результате выводятся 4 последовательных приближённых решения. Видно, что при $|x|\le 1$приближения сходятся к истинному решению с высокой точностью.

Метод малого параметра

Метод малого параметра был предложен А. Пуанкаре в 1892 г. Пусть правая часть уравнения (13) зависит от параметра λ, т.е.

$$y' = f(x, y; \lambda) \ . \qquad (19)$$

При этом известно некоторое частное решение *$y_0(x)$* при некотором значении параметра $\lambda=\lambda_0$. Решение будем искать в виде ряда

$$y(x) = \sum_{n=0}^{\infty} (\lambda - \lambda_0)^n y_n(x) \qquad (20)$$

Подставим (20) в (19) и проведём разложение в ряд Тейлора по степеням $(\lambda - \lambda_0)$. Объединим слагаемые с сомножителями одной и той же степени по $(\lambda - \lambda_0)$ и приравняем их нулю, тогда: $y_0' = f(x, y_0; \lambda_0)$

$$y_1' = f_y(x, y_0; \lambda_0) y_1 + f_\lambda(x, y_0; \lambda_0), \qquad (21)$$

$$y_2' = 2 f_y(x, y_0; \lambda_0) y_2 + f_{yy}(x, y_0; \lambda_0) y_1^2 + f_{\lambda y}(x, y_0; \lambda_0) y_1 + f_{\lambda\lambda}(x, y_0; \lambda_0) \text{ и т.д.}$$

Общий член разложения (21) можно представить в виде:

$$y_n' = \alpha_n(x) y_n(x) + \beta_n(x), \qquad n = 1,\ 2,\ \ldots \qquad (22)$$

т.е. определение коэффициентов разложения (20) сводится к решению линейных уравнений (22).

Рассмотрим пример. Пусть $y' = f(x, y; \lambda) = x^2 + (1+\lambda) y^2$ и $\lambda_0 = -1$. В листинге Data_sheet3.m приведен код программы, которая аналитически решает уравнения (21) для нашего примера. Итог работы программы – аналитические выражения для приближений $y_0(x)$, $y_1(x)$, $y_2(x)$ соответственно.

Метод прогноза-коррекции Адамса

Метод основан на аппроксимации интерполяционными полиномами правых частей ОДУ.

Пусть с помощью любого из методов, рассмотренных выше, вычислено решение заданного дифференциального уравнения в точках x_1, x_2, x_3 (а в точке x_0 решение и так известно – поставлена задача Коши). Полученные значения функции обозначим как y_0, y_1, y_2, y_3, а значения правой части дифференциального уравнения как f_0, f_1, f_2, f_3, где $f_k = f(x_k, y_k)$. Начиная с четвертой точки, на каждом шаге интегрирования дифференциального уравнения вычисления осуществляются по схеме

$$\mathbf{P(EC)^{\{m\}}E}$$

где P – прогноз решения; E – вычисление f (x , y); C – коррекция решения; m – количество итераций коррекции. Схемы такого типа называют *«прогноз-коррекция»*: это подразумевает сначала приблизительное вычисление решение по формуле низкого порядка, а затем уточнение с учетом полученной информации о поведении интегральной кривой.

Иногда в методе Адамса используется схема **PECE** на каждом шаге процесса интегрирования, т.е. осуществляется только одна коррекция. Формулы метода также легко переносятся на решение систем ОДУ первого порядка.

Пример 1. Различные формы записи дифференциальных уравнений первого порядка.

$$\left(\frac{dy}{dx}\right)^2 - 2y\frac{dy}{dx} - y^2\left(e^{2x} - 1\right) = 0$$

$$xdy + (y - cosx)dx = 0$$

$$y' + 2y = e^x$$

Те же уравнения, записанные в нормальной форме.

$$\frac{dy}{dx} = y\left(1 \pm e^x\right) \text{ или } y' = y\left(1 \pm e^x\right)$$

$$y' = \frac{1}{x}\left(\cos x - y\right)$$

$$y' = e^x - 2y$$

Пример 2. Проверка правильности решения дифференциального уравнения первого порядка.

Показажем, что $y(x) = \frac{\sin x}{x}$ есть решение уравнения $xdy + (y - cosx)dx = 0$.

Для этого вычислим dy и подставим его в уравнение:

$$dy = \left(\frac{\cos x}{x} - \frac{\sin x}{x^2}\right)dx,$$

$$xdy + (y - \cos x)dx = x\left(\frac{\cos x}{x} - \frac{\sin x}{x^2}\right)dx + \left(\frac{\sin x}{x} - \cos x\right)dx =$$

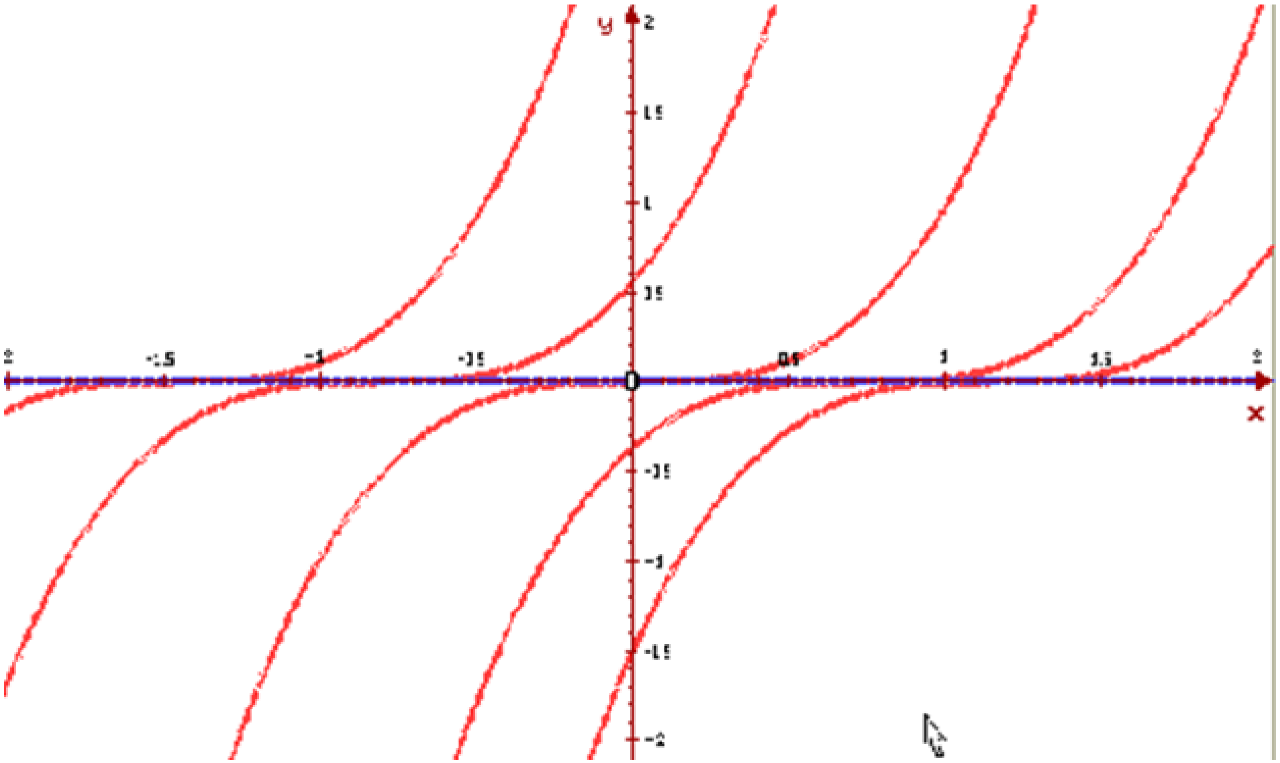
$$= \cos x dx - \frac{\sin x}{x}dx + \frac{\sin x}{x}dx - \cos x dx \equiv 0$$

Пример 3. Пример нарушения единственности решения задачи Коши

Рассмотрим уравнение

$$\frac{dy}{dx} = 3\sqrt[3]{y^2}$$

Его правая часть непрерывна всюду, а частная производная правой части при $y=0$ не существует. На рисунке видно, что через каждую точку с координатами $x = C$, $y = 0$ проходит два решения: $y=0$ и $y = (x - C)^3$.



Пример 4. Решение задачи методом Эйлера.

Применяя метод Эйлера, найти решение задачи Коши: $\begin{cases} y' = y - t \\ y(0) = 1.5 \end{cases}$ в трех последовательных точках $t_1 = 0.2$, $t_2 = 0.4$, $t_3 = 0.6$. Найти точное решение задачи и найти величину абсолютной погрешности в указанных точках.

Возьмем шаг $h = 0.2$. Используя расчетную формулу Эйлера, найдем приближенное решение задачи Коши:

$$y_1 = y_0 + 0.2(y_0 - t_0) = 1.5 + 0.2 \cdot 1.5 = 1.8,$$

$$y_2 = y_1 + 0.2(y_1 - t_1) = 1.8 + 0.2 \cdot (1.8 - 0.2) = 2.12,$$

$$y_3 = y_2 + 0.2(y_2 - t_2) = 2.12 + 0.2 \cdot (2.12 - 0.4) = 2.464.$$

Таким образом, получили численное решение задачи Коши с шагом h:

t_i	0	0.2	0.4	0.6
y_i	1.5	1.8	2.12	2.464

В этой задаче легко находится точное решение, например, методом вариации постоянной:

$y(t) = 0.5e^t + t + 1$. Вычислим значения точного решения в указанных точках.

t_i	0	0.2	0.4	0.6
$y(t_i)$	1.5	1.811	2.146	2.511

Абсолютную погрешность вычислим так: $r_i = |y(t_i) - y_i|$. Тогда $r_1 = 0.011$, $r_2 = 0.026$, $r_3 = 0.047$. Таким образом, максимальная величина погрешности равна $R \approx 0.05$.

Решение дифференциальных уравнений в MATLAB

Для решения дифференциальных уравнений и систем в MATLAB предусмотрены следующие функции **ode45(f, interval, X0 [, options]), ode23(f, interval, X0 [, options]), ode113(f, interval, X0 [, options]), ode15s(f, interval, X0 [, options]), ode23s(f, interval, X0 [, options]), ode23t (f, interval, X0 [,options]) и ode23tb(f, interval, X0 [, options]).**

Входными параметрами этих функций являются:

f - вектор-функция для вычисления правой части уравнения системы уравнений;

interval - массив из двух чисел, определяющий интервал интегрирования дифференциального уравнения или системы;

X0 - вектор начальных условий системы дифференциальных уравнений;

options - параметры управления ходом решения дифференциального уравнения или системы.

Все функции возвращают:

массив T - координаты узлов сетки, в которых ищется решение;

матрицу X, i-й столбец которой является значением вектор-функции решения в узле T_i.

В функции **ode45** реализован метод Рунге-Кутта 4-5 порядка точности, в функции **ode23** также реализован метод Рунге-Кутта, но 2-3 порядка, а функция **ode113** реализует метод Адамса.

Для решения жёстких систем предназначены функция **ode15s**, в которой реализован метод Гира, и функция **ode23s**, реализующая метод Розенброка. Для получения более точного решения жёсткой системы лучше использовать функцию **ode15s.**

Для решения системы с небольшим числом жёсткости можно использовать функцию **ode23t**, а для грубой оценки подобных систем служит функция **ode23tb**.

Символьное решение обыкновенных дифференциальных уравнений произвольного порядка осуществляет функция **dsolve**

r = dsolve('eq1,eq2,...', 'cond1,cond2,...‘,'v')

В листинге Data_sheet1.m приведен код программы, иллюстрирующей использование функции **dsolve** в MATLAB, которая даёт точное решение обыкновенных дифференциальных уравнений произвольного порядка.

MATLAB решил все семь дифференциальных уравнений, при этом последнее уравнение он не смог разрешить относительно зависимой переменной *y*, о чём и доложил.

Data_sheet1.m

```
%примеры аналитического решения обыкновенных
%дифференциальных уравнений с помощью функции
%MATLAB - dsolve
%пример №1 (u'=au)
expl1=dsolve('Du=a*u','x')
%пример №2 (u'=u^2)
expl2=dsolve('Du=u^2','x')
%пример №3 (u''-g=0)
expl3=dsolve('D2u-g=0','x')
%пример №4 (u''+q^2u=0)
expl4=dsolve('D2u+q^2*u=0','x')
%пример №5 (u''-(3/4)x^(-2)u=0)
expl5=dsolve('D2u-(3/4)*x^(-2)*u=0','x')
%пример №6 (u'=f(x))
expl6=dsolve('Du=f(x)','x')
%пример №7 (u'=(u-x)/(u+x))
expl7=dsolve('Du=(u-x)/(u+x)','x')
```

Пример:

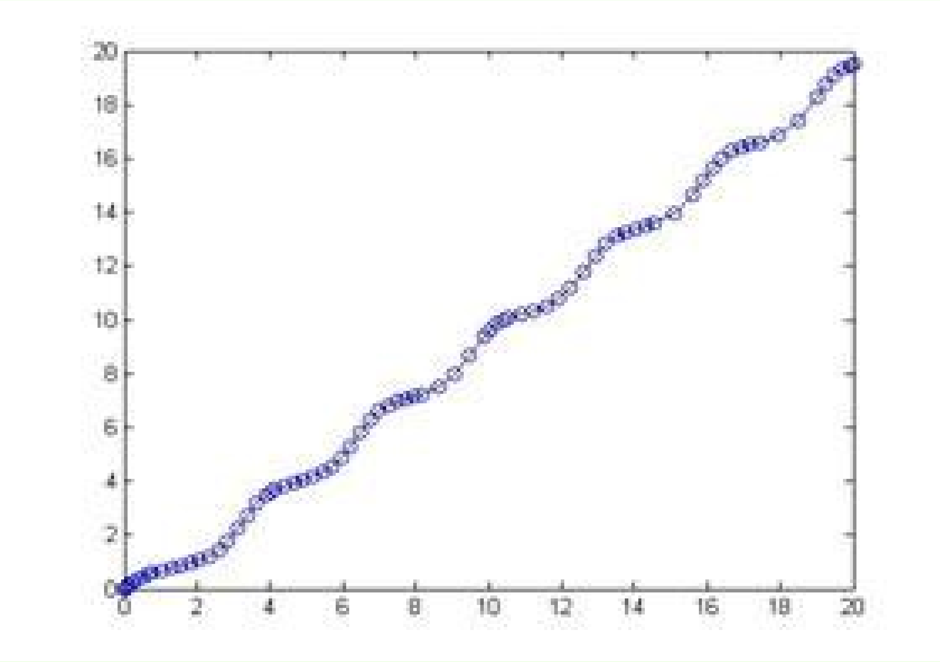
В М-файле с именем pr7.m пишем:

```
function f=pr7(x,y)
f=cos(x+y)+(3/2)*(x-y);
end
```

Потом в командном окне вызываем функцию ode113:

ode113(@pr7,[0 20],0)   %Метод Адамса: @pr7 – ссылка на М-функцию, [0 20]- интервалы интегрирования,0 - условие: y(0)=0

Результатом будет график:



Пример:

Необходимо реализовать метод Рунге-Кутта 4 порядка и решить задачу Коши для предложенной системы дифференциальных уравнений:

$$\frac{d}{dx}y_1 = y_2 \qquad\qquad y_1(0) = 0.1$$

$$\frac{d}{dx}y_2 = \left(\frac{y_1}{x} - y_2\right)\frac{1}{x} - y_1 \qquad\qquad y_2(0) = 0.5$$

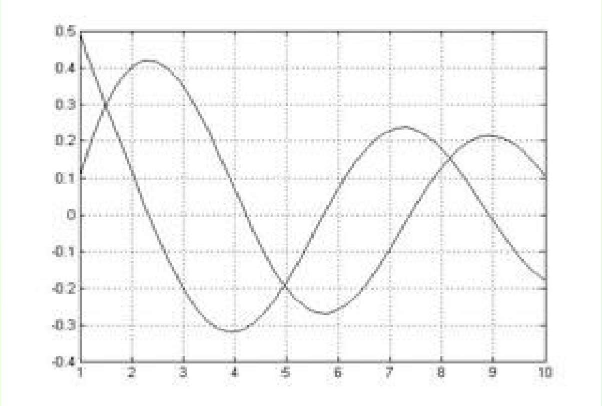
В М-файле с именем pr8.m пишем:

```
function dy=pr8(x,y)
dy=zeros(2,1);
dy(1)=y(2);
dy(2)=((y(1)/x)-y(2))*(1/x)-y(1);
end
```

Потом в командном окне вызываем функцию ode45:

```
[x,y]=ode45(@pr8,[1 10], [0.1 0.5]);
plot(x,y,'-k')
grid;
```

Результатом будет график:



Пример.

```
% Решить задачу Коши методом Эйлера и оценить погрешность.
    % Введём функцию
    f = inline('y-t');
% Начальные условия
    y0 = 1.5;
% Точное решение
    ye = dsolve('Dy=y-t','y(0)=1.5');
% Приближённое решение по методу Эйлера, t=0..1
    n = 100;
    h = 1 / n;
    y = [];
    t0 = 0;
    for i=1:n
    y(end+1) = y0;
    y0 = y0 + h * f(t0, y0);
    t0 = t0 + h;
    end
% Найдём погрешность решения
    t = linspace(0, 1, n);
    dy = max(abs(y - subs(ye, t)))
>>
    dy = 0.0301
```

Задания

Задача 1. Проверить, являются ли данные функции решениями указанных дифференциальных уравнений:

$$y(x) = e^{-2x} + \frac{1}{3}e^{x}, \quad y' + 2y = e^{x}$$

$$y(x) = 2 + \sqrt{1 - x^2}, \quad \left(1 - x^2\right)y' + xy = 2x$$

Задача 2.

А) Найти методом Эйлера на отрезке [0, 1] с шагом h=0.2 и с шагом 0.05 приближенное решение задачи Коши

$$y' = \sin(x) - \cos(y), \quad y(0) = 1.$$

Б) Найти решение этой же задачи методом Рунге-Кутта 4 порядка с шагом h=0.2 и с шагом 0.05.

Изобразить все приближенные решения на одном графике.

Оценить погрешность в каждом случае по Рунге.