# Численные методы решения обыкновенных дифференциальных уравнений

Рассматриваются уравнения первого порядка, разрешимые относительно первой производной с начальными условиями :

 (6.1)

Существует теорема Коши о единственности решения дифференциального уравнения при заданных начальных условиях . Геометрически  определяет поле направлений на плоскости , а решения обыкновенного дифференциального уравнения (ОДУ) - интегральные кривые.

Численные методы решения задачи Коши для ОДУ основаны на том, что решение можно представить в виде разложения в ряд Тейлора с любой степенью точности.

## 6.1 Метод разложения в ряд Тейлора.

Решение ищется в виде



Функциональные зависимости известны:

,

, (6.2)

 и т.д.

Этот метод приводит к громоздким выражениям для производных, и в основном используются для получения других численных методов.

## 6.2. Общая схема метода Рунге - Кутта

Одношаговые методы позволяют получить заданную точность используя только предыдущее значение . Изменение  на шаге *h* представляется в виде квадратурной формулы (типа Гаусса):

,

где .

Для получения коэффициентов , и квадратурная сумма разлагается в ряд по степеням *h.*  Полученное разложение сравнивается с рядом Тейлора:

 (6.2.1)

В общем виде выражения для коэффициентов получить трудно, поэтому рассмотрим наиболее употребительные формулы.

Введем обозначения:

,

, (6.2.2)

,

………………………………………….



Квадратурную формулу разлагаем в ряд по *h*:

 (6.2.3)

ƒ, ƒ, ƒ, ƒ -- частные производные по *x* и *y* ƒ(*x, y*).

Полученное разложение сравнивается с рядом Тейлора (6.2.1).

Рассмотрим несколько частных случаев.

## 6.3 Методы Рунге-Кутта низших порядков

### 6.3.1 Метод Эйлера

В квадратурной формуле ограничиваемся одним слагаемым:

. (6.3.1.1)

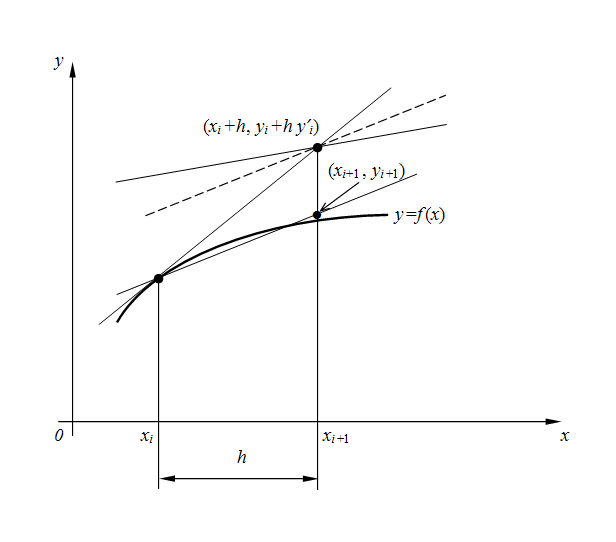
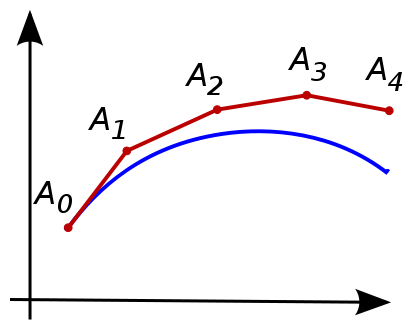
Интегральная кривая заменяется ломаной линией, состоящей из прямолинейных отрезков.

Выбирается шаг  *h* и значение функции в точке  ищется по формуле



т.е. в интегральном уравнении f(x,y) заменяется на константу.

Ошибки метода так как в ряде Тейлора отбрасываются вторые производные.



### 6.3.2. Метод трапеций и прямоугольника

Это популярные методы, иначе их называют метод Коши- Эйлера и модифицированный метод Эйлера, их ошибка 

Представление  - позволяет сравнить два первых слагаемых в разложении с рядом Тейлора:

, , 

Получены три уравнения для четырех неизвестных, что является общим свойством метода Рунге- Кутта. То есть для каждого порядка точности существует множество вычислительных схем: , *.*

Положим , тогда

, (6.3.2.1)

то есть значение производной «подправляется» значением в предварительно определенной точке.

, тогда , 

В этом случае 

image119.gif(702 bytes) - Коши-Эйлера

image126.gif(829 bytes)

- модифицированный метод Эйлера



## 6.4. Методы Рунге-Кутта высших порядков

В методе Рунге- Кутта **третьего порядка** точности:



Разлагая в ряд по *h*  до h и сравнивая с рядом Тейлора (7.2.1 )получим следующую систему из шести уравнений для восьми неизвестных:



Наиболее употребительна в этом случае симметричная разностная схема ( аналог метода парабол при численном интегрировании ): , тогда:

, , , , , .

,

,

,

.

В методе Рунге-Кутта точности порядка  получается система из 11 уравнений для 13 неизвестных.

Наиболее употребительны две вычислительные схемы:

1. Аналог метода 3/8 в численном интегрировании.

, где

 ,

,

,

,

1. Аналог метода парабол.

, (6.4.1)

где ,

,

,

.



Проблема выбора той или иной вычислительной схемы при заданной точности зависит от вида *ƒ(х,у)*, так как от этого зависит величина остаточного члена.

## 6.5. Задание к теме и пример решения ОДУ

Найти решение задачи Коши для ОДУ:

,  на интервале . *K* и *L* параметры из табл.

Решить пятью методами:

1. Метод вариации постоянных (точное решение).

2. Разложение в ряд Тейлора до четвертого порядка.

1. Метод Эйлера (6.3.1.1).
2. Метод трапеций (Коши-Эйлера) (6.3.2.1).
3. Метод Рунге-Кутта (6.4.1).

Построить графики и сравнить точность различных методов, шаг .

1. В методе вариации постоянных решение ищется в виде , Однородное уравнение  имеет очевидное решение . Подстановка в неоднородное уравнение дает уравнение для коэффициента: . После интегрирования и подстановки начального условия получим: .

2. Разложение в ряд Тейлора проводится в точке *х*=0. Все производные в этой точке известны ;

, ;

, ;

, .



3. Метод Эйлера. Расчет ведется по формуле (6.3.1.1):

4. Метод Коши-Эйлера (метод трапеций). Вначале рассчитывается значение , которое затем используется в окончательном выражении (6.3.2.1).

1. Метод Рунге-Кутта. Последовательно вычисляются значения производной в промежуточных точках и используются в окончательном выражении с заданными весами (6.4.1).

Пример. *К*=3, *L=*2. , .

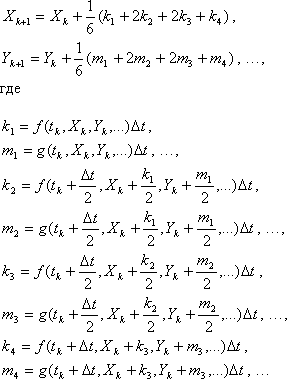
Результаты расчетов представлены в таблице:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| x | 0 | 0,5 | 1 | 1,5 | 2 |
| Точное решение | 2 | 3,34488 | 5,87313 | 10,6768 | 19,5562 |
| Ряд Тейлора | 2 | 3,3438 | 5,83333 | 10,3438 | 18 |
| Метод Эйлера | 2 | 3 | 4,625 | 7,4375 | 12,2812 |
| Метод Коши- Эйлера | 2 | 3,3125 | 5,72656 | 10,2432 | 18,4889 |
| Метод Рунге- Кутта | 2 | 3,34440 | 5,87111 | 10,6710 | 19,5423 |

Метод позволяет решать системы обыкновенных дифференциальных уравнений (ОДУ) первого порядка следующего вида:

127961

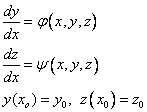
которые имеют решение: 127958



#### Численные методы решения систем ОДУ первого порядка

Рассмотренные методы могут быть использованы также для решения систем дифференциальных уравнений первого порядка.

Покажем это для случая системы двух уравнений первого порядка:



Явный метод Эйлера:

image053

Модифицированный метод Эйлера:

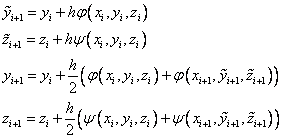
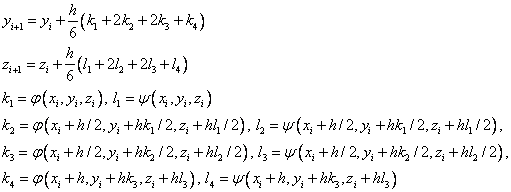


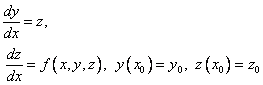
Схема Рунге – Кутта четвертого порядка точности:



К решению систем уравнений ОДУ сводятся также задачи Коши для уравнений высших порядков. Например, рассмотрим **задачу Коши для уравнения второго порядка**

image056

Введем вторую неизвестную функцию image057. Тогда задача Коши заменяется следующей:



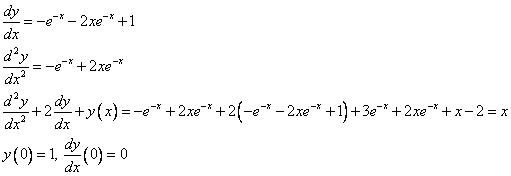
Т.е. в терминах предыдущей задачи: image059.

**Пример. Найти решение задачи Коши**:

image060 на отрезке [0,1].

Точное решение: image061

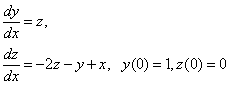
Действительно:



Решим задачу явным методом Эйлера, модифицированным методом Эйлера и Рунге – Кутта с шагом h=0.2.

Введем функцию image063.

Тогда получим следующую задачу Коши для системы двух ОДУ первого порядка:



Явный метод Эйлера:

image065

Модифицированный метод Эйлера:



Метод Рунге – Кутта:

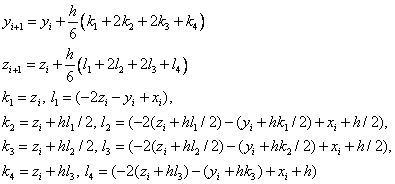


Схема Эйлера:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| X | y | z | y теор | z теор | y-y теор |
| 0 | 1 | 0 | 1 | 0 | 0 |
| 0.2 | 1 | -0.2 | 0.983685 | -0.14622 | 0.016315 |
| 0.4 | 0.96 | -0.28 | 0.947216 | -0.20658 | 0.012784 |
| 0.6 | 0.904 | -0.28 | 0.905009 | -0.20739 | 0.001009 |
| 0.8 | 0.848 | -0.2288 | 0.866913 | -0.16826 | 0.018913 |
| 1 | 0.80224 | -0.14688 | 0.839397 | -0.10364 | 0.037157 |

Модифицированный метод Эйлера:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| X | ycv | zcv | y | z | y теор | z теор | y-y теор |
| 0 | 1 | 0 | 1 | 0 | 1 | 0 | 0 |
| 0.2 | 1 | -0.2 | 1 | -0.18 | 0.983685 | -0.14622 | 0.016315 |
| 0.4 | 0.96 | -0.28 | 0.962 | -0.244 | 0.947216 | -0.20658 | 0.014784 |
| 0.6 | 0.904 | -0.28 | 0.9096 | -0.2314 | 0.905009 | -0.20739 | 0.004591 |
| 0.8 | 0.848 | -0.2288 | 0.85846 | -0.17048 | 0.866913 | -0.16826 | 0.008453 |
| 1 | 0.80224 | -0.14688 | 0.818532 | -0.08127 | 0.839397 | -0.10364 | 0.020865 |

Схема Рунге - Кутта:

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| x | Y | z | k1 | l1 | k2 | l2 | k3 | l3 | k4 | l4 |
| 0 | 1 | 0 | 0 | -1 | -0.1 | -0.7 | -0.07 | -0.75 | -0.15 | -0.486 |
| 0.2 | 0.983667 | -0.1462 | -0.1462 | -0.49127 | -0.19533 | -0.27839 | -0.17404 | -0.31606 | -0.20941 | -0.13004 |
| 0.4 | 0.947189 | -0.20654 | -0.20654 | -0.13411 | -0.21995 | 0.013367 | -0.2052 | -0.01479 | -0.2095 | 0.112847 |
| 0.6 | 0.904977 | -0.20734 | -0.20734 | 0.10971 | -0.19637 | 0.208502 | -0.18649 | 0.187647 | -0.16981 | 0.27195 |
| 0.8 | 0.866881 | -0.16821 | -0.16821 | 0.269542 | -0.14126 | 0.332455 | -0.13497 | 0.317177 | -0.10478 | 0.369665 |
| 1 | 0.839366 | -0.1036 | -0.1036 | 0.367825 | -0.06681 | 0.40462 | -0.06313 | 0.393583 | -0.02488 | 0.423019 |

Версия MathCAD предлагает два подхода к численному решению “хороших” систем. Более старомодный вариант (в стиле ранних версий пакета) позволяет пользователю выбрать одну из встроенных процедур интегрирования и явно обратиться к ней, задав несколько параметров, управляющих ходом вычислительного процесса:

**u:=name fun(y0,t0,t1,M,D).**

Здесь:

**name fun** — имя функции интегрирования (см. табл.1);

**y0** — вектор-столбец начальных значений в момент t0;

**t1** — конец интервала интегрирования (допускается t1 < t0);

**M** — количество точек внутри интервала интегрирования, в которых нужно получить решение;

**D** — векторная функция двух аргументов (независимой переменной t и вектора y), описывающая правые части системы ОДУ.

Функция name fun возвращает матрицу с результатами интегрирования: её первый столбец (**u(0**)) содержит массив значений независимой переменной **t**, остальные столбцы — **u(1), u(2)**... — содержат компоненты искомой вектор-функции **y**.

№ Имя функции Пояснение

1 Rkfixed Метод Рунге–Кутта с фиксированным шагом

2 Rkadapt Метод Рунге–Кутта с переменным шагом

3 Bulstoer Метод Булирша–Штера

Более современный вариант не требует приведения ОДУ к каноническому виду и построен по типу **Given** (Дано)–**Odesolve** (Реши систему ОДУ).

Однако исходное ОДУ должно быть разрешимо относительно старшей производной. Начальный выбор метода и задание управляющих параметров производится автоматически, хотя пользователю предоставляется возможность изменить те или иные управляющие параметры, предусмотренные процедурой **Odesolve**:

**u := Odesolve([vy,] t,t1 [,step]).**

Здесь:

**vy** — вектор искомых функций. Данный параметр можно опустить, если

решается уравнение первого порядка. Если вектор **vy** задан, то результаты

решения системы будут выданы в заданном порядке;

**t** — независимая переменная;

**t1** — конец интервала интегрирования (в этом случае началом интервала по умолчанию считается **t0**=0). Вместо этого может быть задан двухэлементный вектор-столбец начала **t0** и конца **t1** интервала.

**step** — необязательный параметр, задающий количество равномерно отстоящих узлов на интервале интегрирования (по умолчанию **step = 100**), в которых строится решение.