

Министерство образования и науки Российской Федерации
Санкт-Петербургский политехнический университет Петра
Великого

—
Институт компьютерных наук и технологий
Высшая школа программной инженерии

Курсовая работа
по дисциплине «Вычислительная математика»

Выполнил
Студент группы 5130904/20004

Шелковников Д.С.

Преподаватель

Устинов С.М.

Оглавление

Задание	2
Результаты.....	3
Вывод.....	7
Код программы.....	8
<DIR>/computational_mathematics/coursework/main.cpp.....	8

Задание

Решение краевой задачи сведением к задаче Коши.

Для решения нелинейной краевой задачи относительно $y(x)$ на интервале $0 \leq x \leq 1$

$$\frac{d^2 y}{dx^2} = y^2 - 1,$$

$$y(0) = 0,$$

$$y(1) = 1$$

может быть использован следующий подход.

Исходное уравнение переписываем в виде $\frac{d}{dx} \left(\frac{(y')^2}{2} - \frac{y^3}{3} + y \right) = 0$

$$\text{Отсюда } \frac{(y')^2}{2} - \frac{y^3}{3} + y = \alpha \quad (1)$$

где α -некоторая константа. Поскольку $y(0)=0$, то $y'(0) = \sqrt{2\alpha}$.

Если бы мы могли вычислить α , то исходная задача свелась бы к задаче Коши, легко решаемой с помощью подпрограммы RK45.

Интегрирование уравнения (1) дает $x = \int_0^y \frac{dy}{\sqrt{2 \left(\alpha + \frac{y^3}{3} - y \right)}}.$

Используя граничное условие $y(1)=1$, получим уравнение для α : $1 = \int_0^1 \frac{dy}{\sqrt{2 \left(\alpha + \frac{y^3}{3} - y \right)}}.$

которое может быть решено с помощью подпрограмм QUANC8 и ZEROIN.

Реализовать этот подход к решению задачи. Оценить погрешность результата и погрешность, определяемую неточностью в исходных данных.

Результаты

$a = 0.666668$, $b = 300$

Find alpha: 0.9711

Solve value = 0

RKF for diff system:

0	0	1.39363
0.1	0.134378	1.29424
0.2	0.258963	1.19827
0.3	0.374238	1.10845
0.4	0.480929	1.02689
0.5	0.579945	0.95517
0.6	0.672332	0.894509
0.7	0.759247	0.845865
0.8	0.841933	0.810065
0.9	0.921715	0.787912
1	1	0.7803

Play with alpha:

-3%, alpha = 0.941967

RKF for diff system:

0	0	1.37256
0.1	0.132272	1.27316
0.2	0.254743	1.17706
0.3	0.367882	1.0869
0.4	0.47239	1.0047
0.5	0.569139	0.931966
0.6	0.659136	0.869813
0.7	0.743484	0.819115
0.8	0.823365	0.780594
0.9	0.900031	0.754932
1	0.974805	0.742873

-2%, alpha = 0.951678

RKF for diff system:

0	0	1.37962
0.1	0.132977	1.28022
0.2	0.256157	1.18417
0.3	0.370012	1.09412
0.4	0.47525	1.01213
0.5	0.572759	0.939735
0.6	0.663556	0.878079
0.7	0.748763	0.828065
0.8	0.829582	0.790448
0.9	0.90729	0.765952
1	0.983236	0.755369

-1%, $\alpha = 0.961389$

RKF for diff system:

0	0	1.38664
0.1	0.13368	1.28725
0.2	0.257563	1.19124
0.3	0.37213	1.10131
0.4	0.478097	1.01953
0.5	0.57636	0.94747
0.6	0.667954	0.886311
0.7	0.754017	0.836981
0.8	0.835771	0.800271
0.9	0.914517	0.776945
1	0.991634	0.767844

0%, $\alpha = 0.9711$

RKF for diff system:

0	0	1.39363
0.1	0.134378	1.29424
0.2	0.258963	1.19827
0.3	0.374238	1.10845
0.4	0.480929	1.02689
0.5	0.579945	0.95517
0.6	0.672332	0.894509
0.7	0.759247	0.845865
0.8	0.841933	0.810065
0.9	0.921715	0.787912
1	1	0.7803

1%, $\alpha = 0.980811$

RKF for diff system:

0	0	1.40058
0.1	0.135074	1.3012
0.2	0.260356	1.20527
0.3	0.376336	1.11556
0.4	0.483748	1.03421
0.5	0.583511	0.962835
0.6	0.676689	0.902673
0.7	0.764453	0.854717
0.8	0.848068	0.819828
0.9	0.928882	0.798852
1	1.00833	0.792735

2%, $\alpha = 0.990522$

RKF for diff system:

0	0	1.4075
0.1	0.135765	1.30812
0.2	0.261741	1.21224
0.3	0.378423	1.12264
0.4	0.486552	1.0415
0.5	0.587061	0.970467
0.6	0.681025	0.910805
0.7	0.769635	0.863538
0.8	0.854176	0.829562
0.9	0.93602	0.809768
1	1.01663	0.805151

3%, alpha = 1.00023

RKF for diff system:

0	0	1.41438
0.1	0.136454	1.31501
0.2	0.26312	1.21917
0.3	0.3805	1.12969
0.4	0.489344	1.04876
0.5	0.590594	0.978066
0.6	0.685341	0.918904
0.7	0.774794	0.872327
0.8	0.860257	0.839268
0.9	0.943129	0.820658
1	1.02491	0.817548

Вывод

Определим сначала интервалы для поиска решения.

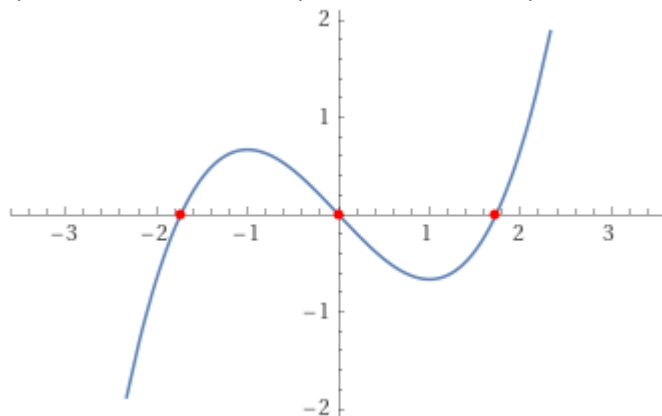


График знаменателя (подкорневого выражения) имеет такой вид. По условию интегрирования нас интересует часть от 0 до 1. Решим неравенство, чтобы часть от 0 до 1 была неотрицательной. Это выполнится при добавлении $\frac{2}{3}$. В таком случае знаменатель будет обращаться в 0, что нам не подходит. Тогда в коде программы укажем минимальную границу для поиска решения, как $2.0/3+0.000001$, а верхнюю без ограничений. Далее с помощью программы ZEROIN найдем решение.

Для решения дифференциального уравнения введем замену $dy = x$ и получим систему из двух дифференциальных уравнений, где начальные параметры: 0 и $\sqrt{2 * \alpha}$, где α мы нашли ранее.

Проанализируем систему дифференциальное уравнение на устойчивость. При изменении α на 1% в большую или меньшую сторону, решение уравнения изменяется на 0.833% в соответствующую

сторону. При изменении α на 2% в большую или меньшую сторону, решение уравнения изменяется на 1.663% в соответствующую сторону. При изменении α на 3% в большую или меньшую сторону, решение уравнения изменяется на 2.491% в соответствующую сторону. Можно сделать вывод, что система не устойчивая, потому что при изменении параметра α , решение изменяется примерно на такое же значение в процентах.

Код программы

```
<DIR>/computational_mathematics/coursework/main.cpp
#include <iostream>
#include <cmath>
#include <functional>
#include "../common/Quanc8.h"
#include "../common/Rkf45.h"
#include "zeroin.h"

double integrand(double y, double alpha) {
    return 1.0 / sqrt(2 * (alpha + pow(y, 3) / 3 - y));
}

int diff_func(int n, double t, double *x, double *dx) {
    dx[0] = x[1];
    dx[1] = x[0] * x[0] - 1;
    return 0;
}

double to_solve(double alpha) {
    double a = 0, b = 1;
    double abserr = 1e-6, relerr = 1e-6;
    auto func = std::bind(integrand, std::placeholders::_1, alpha);
    dimkashelk::Quanc8 quanc8(func, a, b, abserr, relerr);
    return 1 - quanc8.getResult();
}

double find_alpha(double a, double b) {
    double tol = 1e-12;
    int flag = 0;
    double res = zeroin(a, b, to_solve, tol, &flag);
    return res;
}

int main() {
    double a = 2.0 / 3 + 0.000001, b = 300;
    double alpha = find_alpha(a, b);
    std::cout << "a = " << a << ", b = " << b << "\nFind alpha: " << alpha <<
    "\nSolve value = " << to_solve(alpha) << "\n\n";

    std::cout << "RKF for diff system:\n";
    double data[2]{0.0, std::sqrt(2 * alpha)};
    dimkashelk::Rkf45::calculate(diff_func, data, 0, 1);
}
```



```

std::cout << "\n\nPlay with alpha: \n";
for (double i = 0.97; i < 1.04; i += 0.01) {
    double new_alpha = alpha * i;
    std::cout << (i - 1) * 100 << "%, alpha = " << new_alpha << "\n";
    std::cout << "RKF for diff system:\n";
    data[0] = 0.0;
    data[1] = std::sqrt(2 * new_alpha);
    dimkashelk::Rkf45::calculate(diff_func, data, 0, 1);
}
return 0;
}

```

В программе также использовались файлы Quanc8.cpp и Quanc8.h, Rkf45.cpp и Rkf45.h из предыдущих лабораторных работ и файлы zeroin.h, rkf.h из стандартной библиотеки.