|  |  |
| --- | --- |
| Gerb-BMSTU_01 | **Министерство науки и высшего образования Российской Федерации**  **Федеральное государственное бюджетное образовательное учреждение**  **высшего образования**  **«Московский государственный технический университет**  **имени Н.Э. Баумана**  **(национальный исследовательский университет)»**  **(МГТУ им. Н.Э. Баумана)** |

ФАКУЛЬТЕТ *Робототехники и комплексной автоматизации*

КАФЕДРА *Системы автоматизированного проектирования (РК-6)*

**ОТЧЕТ О ВЫПОЛНЕНИИ ЛАБОРАТОРНОЙ РАБОТЫ**

по дисциплине:

Методы математического моделирования

сложных процессов и систем

|  |  |  |
| --- | --- | --- |
| Студент |  | Абидоков Рашид Ширамбиевич |
| Группа |  | РК6-11М |
| Тип задания |  | лабораторная работа |
| Тема лабораторной работы |  | Применение библиотек динамической компоновки для разработки программных реализаций вычислительных методов |

Студент **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Абидоков Р. Ш.**

*подпись, дата фамилия, и.о.*

Преподаватель **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Соколов А. П.**

*подпись, дата фамилия, и.о.*

Оценка \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

*Москва, 2020 г.*

Оглавление

[Задание на лабораторную работу 3](#_Toc52739933)

[Цель выполнения лабораторной работы 3](#_Toc52739934)

[Выполненные задачи 3](#_Toc52739935)

[1. Аналитическое решение 4](#_Toc52739936)

[2. Описание программной реализации 4](#_Toc52739937)

[3. Графическое сравнение решений 6](#_Toc52739938)

[4. Вычисление максимально допустимых шагов интегрирования 7](#_Toc52739939)

[5. Вариант реализации решения задачи с интервально заданными  
 параметрами 8](#_Toc52739940)

[Заключение 9](#_Toc52739941)

# Задание на лабораторную работу

Тело, имеющее в начальный момент времени , поместили в среду, температура которой поддерживается неизменной и равна .

Экспериментально установлено, что при определенных упрощениях скорость изменения температуры тела пропорциональна разности температур тела и окружающей среды.

Требуемые для реализации численные методы: метод Эйлера, метод Хъюна.

# Цель выполнения лабораторной работы

**Цель выполнения лабораторной работы** – получение практических навыков разработки и применения библиотек динамической компоновки на примере различных реализаций задачи численного решения ОДУ.

# Выполненные задачи

1. По имеющейся математической модели в форме ОДУ получено аналитическое решение
2. На языке C++ написана программа, реализующая возможность численного решения ОДУ с использованием динамически подключаемых библиотек, реализующих методы Ньютона и Хойна. Обеспечена возможность подключения дополнительных методов решения
3. Проведено графическое сравнение методов Ньютона и Хойна с аналитическим решением
4. Вычислены максимально допустимые шаги интегрирования по времени для примененных численных методов
5. Предложен вариант доработки алгоритма для решения с заданием одного из параметров модели в интервальном виде

# Аналитическое решение

Модель процесса представлена в виде ОДУ

где некоторый коэффициент пропорциональности,

температура тела в момент времени

температура окружающей среды

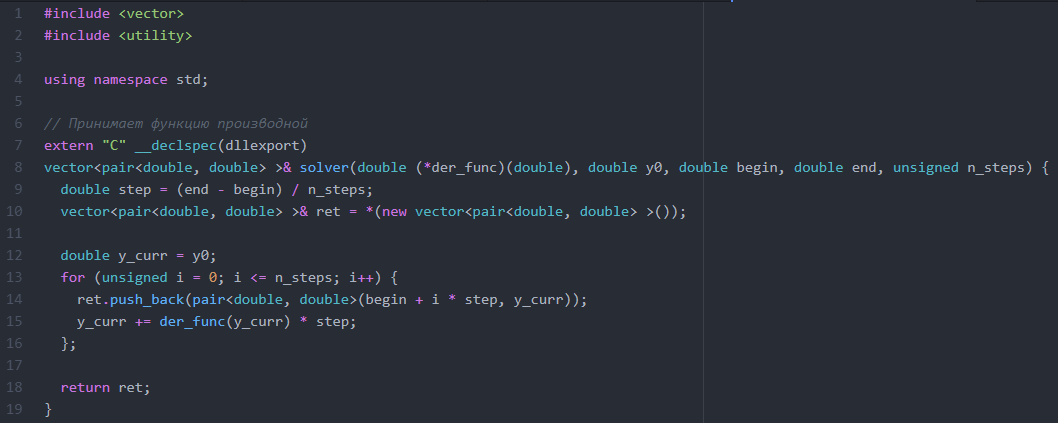
Аналитическим решением данного уравнения является

где температура тела в начальный момент времени

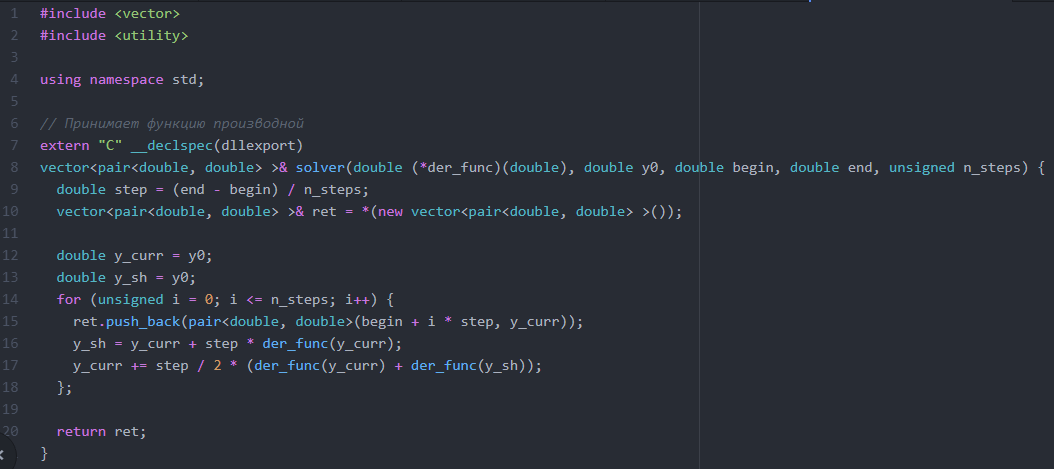
# Описание программной реализации

Исходный код программы разделен на файлы main.cpp, euler.cpp, heun.cpp, config.hpp и config.cpp. Файлы euler.cpp и heun.cpp являются реализациями методов Эйлера и Хойна соответственно и компилируются в динамические библиотеки euler.dll, heun.dll. Их сходный код приведен в Листингах 1 и 2.

Листинг 1. Метод Эйлера



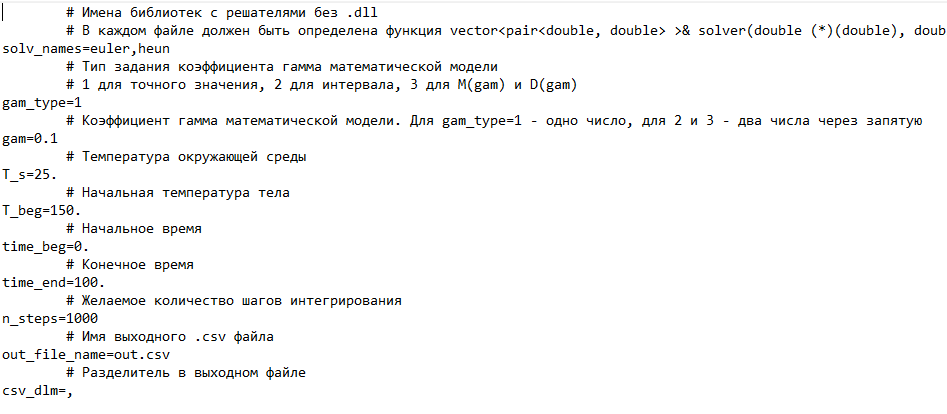
Листинг 2. Метод Хойна



В main.cpp описана основная логика программы – чтение конфигурации из config.txt (в случае его отсутствия, создается файл по умолчанию), вызов решателей из подключаемых библиотек, запись полученных численных и аналитического результатов в выходной .csv-файл.

В свою очередь, config.hpp и config.cpp являются статической библиотекой, реализующей функционал работы с config-файлом.

Листинг 3. Структура config.txt



В config.txt задаются имена используемых файлов решателей, параметры математической модели, имя выходного файла и его разделитель.

# Графическое сравнение решений

Графики построены с использованием библиотеки matplotlib языка python. Здесь и в дальнейшем интервал интегрирования секунд. Приведенные графики построены с количеством шагов 100, шаг интегрирования 1 с для наглядной разницы между решениями. Остальные параметры по умолчанию (см. Листинг 3)

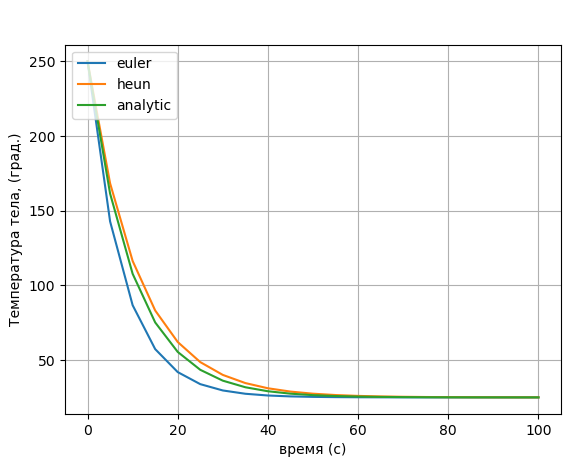


Рис 1.

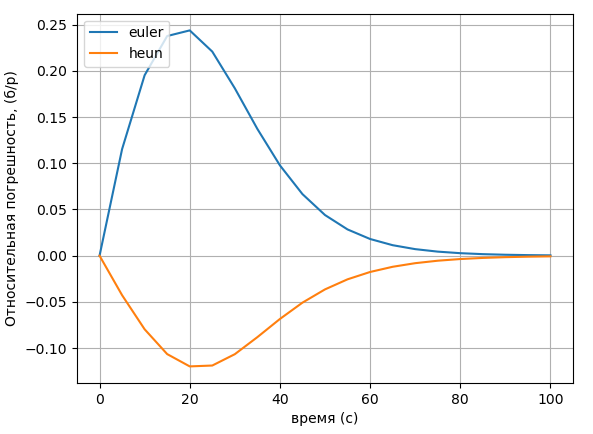


Рис 2.

Как можно видеть, метод Хойна обеспечивает лучшую точность, нежели метод Эйлера.

# Вычисление максимально допустимых шагов интегрирования

# Расчеты проводились при помощи скриптов на языке python с использованием библиотек numpy и pandas. Интервал интегрирования везде одинаковый, изменялось количество шагов интегрирования. Для каждого случая вычислялась величина шага и следующая метрика, основанных на L2-норме:

т.е., фактически, значение Евклидовой длины вектора относительных ошибок, разделенное на размерность этого вектора.

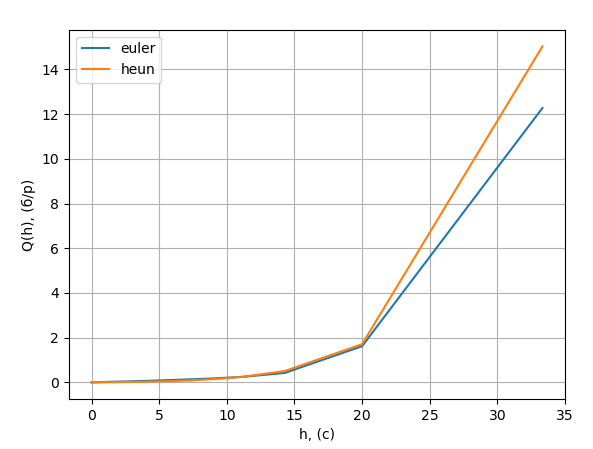


Рис 3. Большой масштаб

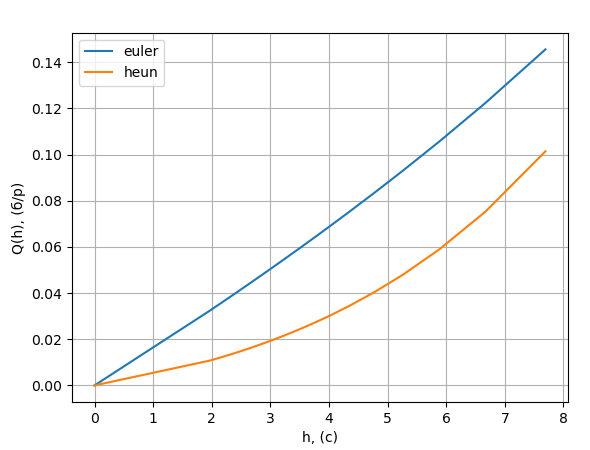


Рис 3. Малый масштаб

Откуда видно, что для обеспечения точности порядка 5%, максимально допустимыми шагами интегрирования являются 3 секунды для метода Эйлера и порядка 5.5 секунд для метода Хойна соответственно.

# Вариант реализации решения задачи с интервально заданными параметрами

Данный раздел является размышлениями на тему "что можно было бы сделать, получив на вход параметр, заданный в интервальном виде либо матожиданием и дисперсией".

*Один из параметров задан интервалом.* Например, . В этом случае интерес может представлять интервал значений некоторой выходной функции решение исходного уравнения. Т.е. фактически задача будет представлять собой поиск и , в простейшем случае просто минимальное и максимальное возможные значения .

Здесь возможно применение методов оптимизации, к примеру, градиентных. Саму производную можно было бы вычислить численно, если предположить, что уравнение может быть интерполировано как по , так и по Задаемся некоторым начальным приближением , численно решаем уравнение с различными вокруг (кол-во зависит от выбранного выражения производной), находим градиент искомой ф-ии, сдвигаемся и т.д. до тех пор, пока не найдем min/max либо не упремся в границу.

*Один из параметров задан математическим ожиданием и дисперсией.* Т.е. задано распределение вероятности параметра (предполагается, что оно имеет вид нормального распределения). В этом случае может представлять интерес распределение вероятности некоторой выходной функции .

В этом случае можно было бы выбрать некоторый интервал, например,   
 и, равномерно разбив его на участки, для каждого найти численное решение и значение искомой функции

И получить таким образом распределение (предположив его вид, например, также нормальное распределение), если бы имела постоянную вероятность внутри выбранного интервала. Т.е. фактически у нас имелось бы правило, по которому можно было преобразовать исходное распределение и получить итоговое, но для предложения конкретного алгоритма вопрос требуется изучить более подробно.

# Заключение

Реализована программа, осуществляющая численное решение ОДУ и сравнение полученных решений с аналитическим. Показано, что метод Хойна обеспечивает лучшую по сравнению с методом Эйлера точность. Получен практический опыт использования динамических библиотек позднего связывания.

Найдены предельно допустимые шаги интегрирования по времени, составляющие 3 секунды для метода Эйлера и порядка 5.5 секунд для метода Хойна соответственно.

Приведены возможные способы решения задачи с интервально заданным параметром.