УДК 519.62; 519.63

О ПРОБЛЕМЕ РЕАЛИЗАЦИИ МАТЕМАТИЧЕСКИХ МЕТОДОВ ПРИБЛИЖЕННОГО РЕШЕНИЯ ДИФФЕРЕНЦИАЛЬНЫХ УРАВНЕНИЙ В ПРОГРАММНЫХ СРЕДАХ

М. А. Дмитриев

Елецкий государственный университет имени И. А. Бунина, Елец, Россия, demin.mishka@yandex.ru

Статья посвящена некоторым аспектам проблемы решения дифференциальных уравнений с использованием численных методов и программных сред. Рассмотрены ключевые вопросы этой проблемы, такие как вычислительная сложность и устойчивость методов. Особое внимание уделено методам решения дифференциальных уравнений первого порядка: Эйлера, Рунге-Кутта, конечных разностей. Представлен краткий обзор некоторых программных сред.

Ключевые слова: дифференциальные уравнения; численные методы; приближённое решение; прикладные задачи; программные среды; Python; MATLAB; Mathcad.

ON THE PROBLEM OF IMPLEMENTING MATHEMATICAL METHODS FOR APPROXIMATE SOLUTIONS OF DIFFERENTIAL EQUATIONS IN SOFTWARE ENVIRONMENTS

M. A. Dmitriev

Bunin Yelets State University, Yelets, Russia, demin.mishka@yandex.ru

The article is devoted to some aspects of the problem of solving differential equations using numerical methods and software environments. Key issues of this problem, such as computational complexity and stability of methods, are considered. Particular attention is paid to methods for solving first-order differential equations: Euler, Runge-Kutta, finite differences. A brief overview of some software environments is presented.

***Keywords*:** differential equations; numerical methods; approximate solution; applied problems; software environments; Python; MATLAB; Mathcad.

Введение

Дифференциальные уравнения являются фундаментальным инструментом математического моделирования в естественных науках, инженерии, экономике и других областях. Они позволяют описывать многие динамические процессы, однако для большинства реальных задач процесс получения аналитического решения является весьма затруднительным. Поэтому численные методы и их программная реализация становятся едва ли не ключевыми для выполнения расчётов в современных исследованиях.

Методология исследования

В области дифференциальных уравнений имеется множество различных проблем, важнейшей из которых, на наш взгляд, является сложность аналитического решения. Для многих из них получение точного аналитического решения не представляется возможным в силу разных причин (например, из-за отсутствия точного алгоритма решения для некоторых классов уравнений; либо из-за наличия структурных особенностей в записи самого дифференциального уравнения, например, наличие специальных функций и т.п.). Следует также учесть вычислительную сложность получения ответа. Для решения многих систем дифференциальных уравнений, зачастую, требуются значительные временные затраты и трудоёмкие расчёты, что повышает вероятность появления ошибки.

При использовании приближённых методов (метод Пикара, интегрирование с помощью степенных рядов, метод ломаных Эйлера, метод Рунге Кутта, метод Адамса, метод Милна) [1] также стоит обратить внимание на проблему точности и устойчивости используемых численных методов. Помимо всего прочего, необходимо моделировать реальные процессы, то есть учитывать начальные и / или граничные условия, а также адаптировать модели под конкретные задачи (например, из биологии, экологии, физики). Отметим лишь, что при решении краевых задач для ОДУ используют: метод коллокации, метод конечных разностей, метод Галёркина, метод Галёркина-Бубнова, метод конечных элементов [2].

Все эти факторы обуславливают необходимость применения программных сред, которые находят свою практическую реализацию и помогают ускорить решение разнообразных проблем во многих областях.

Результаты

Методы приближенного решения дифференциальных уравнений

Дифференциальные уравнения могут решаться различными методами, такими, как:

1. Метод Эйлера (он является простейшим из перечисленных, но имеет низкую точность);

2. Методы Рунге-Кутта (имеют высокую точность и широко используются для решения дифференциальных уравнений);

3. Метод конечных разностей (используется для уравнений в частных производных);

4. Метод конечных элементов (для сложных геометрий) и т.д.

Однако ключевые проблемы решения дифференциальных уравнений напрямую определяют выбор методов их анализа и каждый математический метод требует специфической программной инфраструктуры.

Сравнение различных математических систем

Для анализа дальнейшей работы математических систем изучен фрагмент кода каждой из использованных систем (Pyhton, Mathcad).

Например, требуется решить дифференциальное уравнение первого порядка

$$\frac{dy}{dt} = -2y,$$

где y — это функция, которую мы хотим найти, а t — это независимая переменная, по отношению к которой мы дифференцируем. Функция dsolve в SymPy используется для решения дифференциальных уравнений [3].

«Метод Рунге-Кутты четвёртого порядка — один из самых популярных численных методов. В отличие от того же метода второго порядка, он повышает точность за счёт четырёх вычислений скорости изменения на каждом временном шаге» [4].

«Методы Эйлера и Рунге-Кутты позволяют решать дифференциальные уравнения, когда аналитическое решение недоступно. Однако они могут накапливать ошибки, особенно при больших шагах или длительном моделировании. Поэтому важно проверять точность решения и подбирать оптимальный метод и шаг интегрирования» [3].

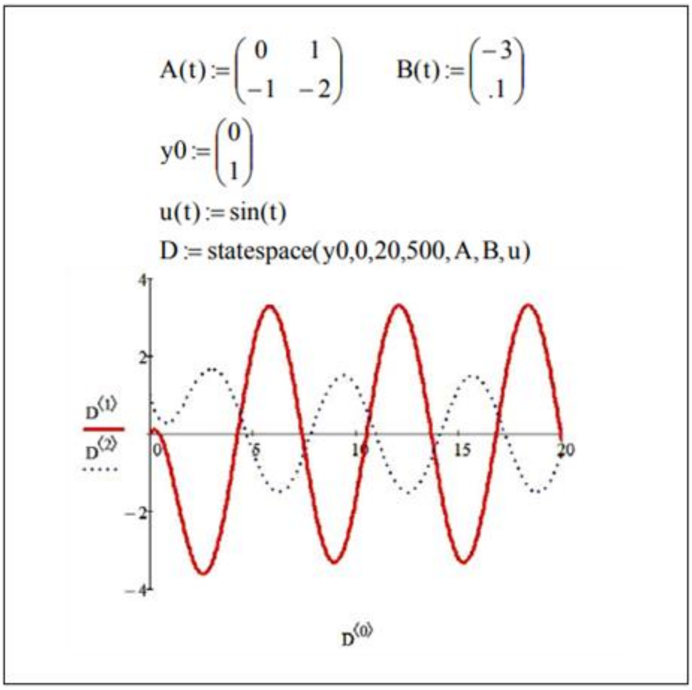
Преимуществами Python являются: открытый и бесплатный инструментарий, богатая экосистема научных библиотек, а также гибкость в настройке параметров. Одним из его недостатков является низкая производительность, по сравнению с компилируемыми языками.

«Разработчики Mathcad 14 предусмотрели дополнительную встроенную функцию statespace решения линейных ОДУ определенного вида

$$y'(t) = A(t)y(t) + B(t)u(t),$$

где: statespace (y_0, t_0, t_1, M, A, B,u) – функция решения систем линейных ОДУ; y_0 – вектор начальных значений в точке t_0;t_0,t_1 – начальная и конечная точки расчета; M – число шагов численного метода; acc – погрешность вычисления; $A(t)$ – матричная функция размера $N \times N$; $B(t)$ – векторная функция размера $N \times k$; $u(t)$ – функция (скалярная, если $k = 1$ или векторная размера $k \times 1$)» [4].

Пример использования функции statespace представлен на рисунке ниже.



Решение системы линейных ОДУ

Таким образом, выбор разнообразных инструментов зависит от многих факторов:

1. Для научных исследований и разработки новых алгоритмов оптимален Python с его богатой экосистемой библиотек.

2. Для инженерных расчётов «из коробки» лучше подходит MATLAB с его специализированными «тулбоксами».

3. Mathcad удобен благодаря интуитивному интерфейсу и символьным вычислениям, но ограничен в возможностях программирования и обработки больших данных по сравнению с другими математическими пакетами.

Конечно, при полноценной реализации того или иного метода, можно получить более детальное представление о плюсах и минусах той или иной среды, имея полный код. Он, естественно, отличается от написанного выше, ввиду необходимости учёта дополнительной информации и реализации в конкретной среде.

Заключение

Численные методы и современные программные среды открывают широкие возможности для решения сложных дифференциальных уравнений, с которыми не справляются аналитические подходы. Представленные методы и их реализации образуют мощный инструментарий для решения актуальных прикладных задач математического моделирования в различных областях науки и техники.

Библиографические ссылки

1. *Левова Г.А., Снежкина О.В.* Математика. Приближенное решение обыкновенных дифференциальных уравнений: учебно-методическое пособие. Пенза: ПГУАС. 2015. 68 с.

2. *Вержбитский В.М.* Основы численных методов: Учебник для вузов. М.: Высшая школа. 2002. 840 с.

3. Решение дифференциальных уравнений с Python [Online] URL: https://habr.com/ru/companies/otus/articles/748532/ (дата обращения: 23.03.2025)

4. *Вельмисов, П.А.* Дифференциальные уравнения в Mathcad: учебное пособие. Ульяновск: УлГТУ. 2016. 109 с.