ISSN 2079-8784
Свидетельство о регистрации СМИ
ЭЛ №ФС77-58104 от 04.10.2014

Вестник ЦЭМИ 2013-2021

ISSN 2079-8784

2

URL - <http://ras.jes.su>

2021

Все права защищены

Выпуск 2 Том . 2021

Решение обыкновенных дифференциальных уравнений в EXCEL с помощью квадратурных и интерполяционных формул

Смоляк Сергей Абрамович

Главный научный сотрудник, Центральный экономико-математический институт РАН Москва, РФ, Нахимовский проспект, 47

Аннотация

Предлагаются два простых и достаточно точных метода численного решения задачи Коши для обыкновенных дифференциальных уравнений, приспособленные для использования в EXCEL и других электронных таблицах.

Ключевые слова: обыкновенные дифференциальные уравнения, численное решение, электронные таблицы

Дата публикации: 26.07.2021

Ссылка для цитирования:

Смоляк С. А. Решение обыкновенных дифференциальных уравнений в EXCEL с помощью квадратурных и интерполяционных формул // Вестник ЦЭМИ – 2021. – Выпуск 2 [Электронный ресурс]. URL: <https://cemi.jes.su/S265838870015151-9-1> (дата обращения: 26.07.2021). DOI: 10.33276/S265838870015151-9

1. Введение

В этой статье речь пойдёт о численном решении задачи Коши для системы обыкновенных дифференциальных уравнений (ОДУ). Для определённости мы будем говорить о нахождении значений вектор-функции $y(t)$ на отрезке $[0, a]$, удовлетворяющей условиям:

$$y'(t) = f(t, y(t)), \quad y(0) = y_0. \quad (1)$$

Вектор-функция f при этом предполагается достаточно гладкой.

Известные методы решения этой задачи, например, метод Рунге-Кутты и другие, описанные, например, в (Бахвалов, 1973; Меркулова, Михайлов, 2014) были разработаны применительно к ситуации, когда вычисления значений функции f представляли определённую сложность, и требовалось минимизировать количество этих вычислений.

Для решения этой задачи имеется много методов, описанных, например, в (Бахвалов и др., 2003; Меркулова, Михайлов, 2014). Наиболее распространённым из них, пожалуй, можно считать метод Рунге-Кутты. Однако такие методы разрабатывались вначале применительно к «ручному счёту», а позднее – применительно к вычислительной технике малой производительности. При этом во многих важных для практических приложений задачах даже вычисление одного из значений функции f представляло определённую сложность. По этой причине указанные методы были ориентированы на минимизацию количества вычислений, необходимых для обеспечения заданной точности решения.

В настоящее время с появлением персональных компьютеров (ПК) ситуация изменилась. С одной стороны, круг объектов, поведение которых моделируется с помощью ОДУ, расширился, а для описания поведения одного и того же объекта используются различные ОДУ. Порой функция f зависит от скалярного или векторного параметра, значения которого требуется подобрать. В таких случаях возникает необходимость решать на ПК уравнение (1) с разными правыми частями. Чаще всего в этих целях используются электронные таблицы, в основном EXCEL. Однако стандартных программ для решения ОДУ в EXCEL нет. С другой стороны, вычисление значений каких-либо аналитически заданных функций, изменение дробления отрезка $[0, a]$ на шаги и проведение итераций в EXCEL не представляет каких-либо сложностей. Однако при записи сложных формул в ячейки электронной таблицы нередко возникают ошибки, которые порой трудно обнаружить. В такой ситуации предпочтение должно отдаваться возможно более простым алгоритмам. Ниже предлагаются два метода, использующие такие алгоритмы.

В них отрезок $[0, a]$ разбивается на N шагов длительности $h = a/N$ точками $t_i = ih$ ($i = 0, 1, \dots, N$), и отыскиваются значения $y_i = y(t_i)$, при которых возможно более точно удовлетворяются вытекающие из (1) условия:

$$y_i = y(t_i) = y_0 + \int_0^{t_i} f(t, y(t)) dt, \quad (i = 1, \dots, N). \quad (2)$$

Для большей наглядности далее будет рассмотрен только одномерный случай (y и f – скаляры) – все приводимые формулы в общем случае сохраняются. Двумерный случай мы рассматриваем на численном примере.

2. Первый метод

Для приближённого вычисления интегралов в (2) можно было бы использовать какую-либо известную и достаточно простую квадратурную формулу. Пожалуй, наиболее подходящей

здесь будет формула Симпсона, связывающая значения интеграла по отрезку со значениями подынтегральной функции в **трёх** точках – двух концах и середине отрезка:

$$\int_b^{b+2h} F(t)dt \approx \frac{h}{3} [F(b) + 4F(b+h) + F(b+2h)]. \quad (3)$$

Если считать подынтегральную функцию достаточно гладкой, то ошибка этой формулы будет иметь порядок h^5 .

Тогда при $i > 1$ равенства (2) можно было бы заменить следующими приближёнными равенствами:

$$y_i = y_{i-2} + \frac{h}{3} [f(t_{i-2}, y_{i-2}) + 4f(t_{i-1}, y_{i-1}) + f(t_i, y_i)]. \quad (4)$$

Если бы такие равенства были справедливы при всех i , это позволило бы определить значения всех y_i с ошибкой порядка h^4 . Однако для $i = 1$ формула (4) не «работает». Казалось бы, здесь нужно воспользоваться другой формулой, связывающей интеграл по отрезку $[0, h]$ со значениями $y_0, f(x_0, y_0), f(x_1, y_1)$, однако любые такие формулы дают ошибку порядка h^3 .

Чтобы исправить положение, предлагается использовать квадратурную формулу, дающую ошибку порядка h^4 , но использующую значения подынтегральной функции на *продолжении отрезка*:

$$\int_0^h F(t)dt \approx \frac{h}{12} [5F(0) + 8F(h) - F(2h)].$$

При таком методе для определения значений функции $y(t)$ в точках t_i используется следующая система соотношений:

$$\begin{aligned} y_1 &= y_0 + \frac{h}{12} [5f(0, y_0) + 8f(t_1, y_1) - f(t_2, y_2)], \\ y_i &= y_{i-2} + \frac{h}{3} [f(t_{i-2}, y_{i-2}) + 4f(t_{i-1}, y_{i-1}) + f(t_i, y_i)], \quad (i = 2, 3, \dots, N). \end{aligned} \quad (5)$$

Для её решения в EXCEL вначале формируются три массива с элементами соответственно y_i, f_i и z_i ($i = 0, 1, \dots, N$).

Первый массив вначале заполняется какими-то начальными приближёнными значениями $y_i = y(t_i)$. Например, можно положить все $y_i = y_0$. Другие способы выбора начального приближения указаны в (Мышенков, Мышенков, 2005).

Во второй массив записываются формулы для вычисления $f_i = f(t_i, y_i)$ по значениям y_i из первого массива.

Третий массив заполняется формулами (5) для вычисления новых значений переменных y_i ($i > 0$) по ранее определённым значениям y_i и $f(t_i, y_i)$.

После этого все y_i заменяются соответствующим значением $z(i)$. (в каждую ячейку первого массива записывается ссылка на соответствующую ячейку третьего массива).

Изложенный метод, по сути, реализует последовательные итеративные расчеты по формулам:

$$\begin{aligned} z_1 &= y_0 + \frac{h}{12}[5f(0, y_0) + 8f(t_1, y_1) - f(t_2, y_2)]; \\ z_i &= y_{i-2} + \frac{h}{3}[f(t_{i-2}, y_{i-2}) + 4f(t_{i-1}, y_{i-1}) + f(t_i, y_i)], \quad (i \geq 2); \\ y_i &= z_i, \quad (i \geq 1). \end{aligned} \quad (6)$$

Система (6) при этом решается в EXCEL автоматически путем выполнения итераций (их количество можно заранее ограничить). При не слишком больших шагах h итерационный процесс сходится, а решение получается достаточно точным. Используя метод, изложенный в (Пименов, 2014), можно показать, что ошибка в значениях y_i будет порядка h^4 . Для контроля точности, как это обычно и делается, проводится аналогичный расчёт с уменьшенным вдвое шагом h .

Пример 1. Уравнение $y' = 2y/(2,5 - t)$, $y(0) = 1$ имеет решением $y(t) = (1 - 0,4t)^{-2}$. Будем решать его предложенным методом на отрезке $[0, 2]$ с шагами разной длительности. Рассчитанные и точные значения $y(t)$ представлены в следующей табл. 1.

Таблица 1. Рассчитанные и точные значения $y(t)$

t	Точные	Длительность шага h				
		0,01	0,02	0,05	0,1	0,2
0	1	1	1	1	1	1
0,2	1,1815	1,1815	1,1815	1,1815	1,1815	1,1811
0,4	1,4172	1,4172	1,4172	1,4172	1,4172	1,4172
0,6	1,7313	1,7313	1,7313	1,7313	1,7313	1,7310
0,8	2,1626	2,1626	2,1626	2,1626	2,1626	2,1627
1	2,7778	2,7778	2,7778	2,7778	2,7778	2,7779
1,2	3,6982	3,6982	3,6982	3,6982	3,6983	3,6994
1,4	5,1653	5,1653	5,1653	5,1653	5,1656	5,1690
1,6	7,7160	7,7160	7,7161	7,7161	7,7171	7,7303
1,8	12,7551	12,7551	12,7551	12,7554	12,7599	12,8180
2	25,0000	25,0000	25,0001	25,0024	25,0343	25,4204

Данный метод, в некотором смысле, близок к одному из неявных методов Адамса-Моултона (Меркулова, Михайлов, 2014, п. 11.6.2; Пименов, 2014; Atkinson, 2009), предназначенных для последовательного «безитерационного» нахождения значений y_i . В нашем же варианте все y_i определяются итеративно и одновременно.

При недостаточно малом шаге h итеративной процесс может не сходиться, а в соответствующих ячейках таблицы появляются, например, чрезмерно большие числа. Выйти из положения можно, опираясь на следующие соображения.

Выберем некоторое число p , лежащее между 0 и 1, и заменим последнюю из формул (6) на другую:

$$y_i = pz_i + (1 - p)y_i, \quad (i = 1, 2, \dots, N). \quad (7)$$

Тогда новыми значениями y_i становятся не только что вычисленные значения z_i , а средние взвешенные из ранее найденных значений y_i и новых значений z_i . Такой алгоритм (за счет увеличения количества итераций, что особой роли не играет) может обеспечить сходимость и при сравнительно больших h . Поэтому при появлении чрезмерно больших y_i достаточно будет только уменьшить p .

Разумеется, предложенный метод применим и при решении задачи Коши для систем ОДУ. В этом случае формулы (6) сохраняются, но переменные y и f надо рассматривать как вектора соответствующей размерности.

Пример 2. Решим на отрезке $[0, 1]$ систему уравнений:

$$\begin{cases} y' = \pi z/2, & y(0) = 0, \\ z' = -\pi y/2, & z(0) = 1. \end{cases}$$

Точным решением этой системы будет $y(t) = \sin \pi t/2$, $z(t) = \cos \pi t/2$. Результаты численного решения при разных шагах h представлены в табл. 2.

Таблица 2. Рассчитанные и точные значения $y(t)$ и $z(t)$

t	Точные		Длительность шага h							
			0,01		0,02		0,05		0,1	
	y	z	y	z	y	z	y	z	y	z
0	0	1	0	1	0	1	0	1	0	1
0,1	0,1564	0,9877	0,1546	0,9880	0,1517	0,9883	0,1470	0,9892	0,1328	0,9899
0,2	0,3090	0,9511	0,3072	0,9517	0,3054	0,9524	0,3000	0,9542	0,2908	0,9570
0,3	0,4540	0,8910	0,4523	0,8920	0,4497	0,8928	0,4455	0,8958	0,4322	0,8994
0,4	0,5878	0,8090	0,5863	0,8103	0,5848	0,8116	0,5801	0,8154	0,5722	0,8213
0,5	0,7071	0,7071	0,7058	0,7087	0,7036	0,7100	0,7005	0,7148	0,6890	0,7209
0,6	0,8090	0,5878	0,8080	0,5896	0,8069	0,5914	0,8037	0,5967	0,7979	0,6052
0,7	0,8910	0,4540	0,8902	0,4560	0,8886	0,4576	0,8870	0,4638	0,8782	0,4717
0,8	0,9511	0,3090	0,9506	0,3111	0,9501	0,3133	0,9486	0,3196	0,9457	0,3300
0,9	0,9877	0,1564	0,9876	0,1587	0,9866	0,1604	0,9868	0,1675	0,9812	0,1763
1	1,0000	0,0000	1,0002	0,0023	1,0004	0,0045	1,0008	0,0113	1,0011	0,0226

Нельзя не отметить и определённого недостатка предлагаемого метода. Известные методы решения ОДУ (скажем, «обычный» метод Рунге-Кутты) допускают выполнение расчётов с

автоматически меняющимся шагом. При данном методе это сопряжено с определенными трудностями. Некоторым оправданием может служить то обстоятельство, что обычно требуется установить значения неизвестной функции $y(t)$ в заранее заданных равноотстоящих друг от друга точках. Данный метод как раз приспособлен именно для этих целей.

3. Второй метод

Основная проблема, решаемая в первом методе, была связана с необходимостью более точной оценки значения функции f в точке (t_1, y_1) , т.е. на начальном шаге алгоритма. Для её решения использовалось значение функции f в точке (t_2, y_2) , т.е. на следующем шаге. Между тем, эту проблему можно решить и иначе, если предварительно найти значение $w_1 = y(h/2)$, пусть и с меньшей точностью. Реализовать эту идею можно следующим образом.

Допустим, что значения y_1 и $f(t_1, y_1)$ известны точно. Тогда о функции $y(t)$ нам будут известны не только её значения в точках 0 и h , но и значения её производной в этих точках $y'(0) = f(0, y_0) = f_0$, $y'(h) = f(h, y_1) = f_1$. Построим теперь интерполяционный многочлен третьей степени $Y(t)$ с такими же значениями функции и производной. Поскольку функция $y(t)$ достаточно гладкая, на отрезке $[0, h]$, ее отклонение от $Y(t)$ будет порядка h^4 : $y(h) = Y(h) + O(h^4)$.

Если теперь выписать уравнение $Y(t)$ в явном виде и подставить в него $t = h/2$, мы получим:

$$w_1 = Y(h/2) = \frac{y_0 + y_2}{2} + \frac{f_0 - f_2}{8} h; \quad y(h/2) = w_1 + O(h^4).$$

Теперь, зная y_0 , y_1 , w_1 , f_0 , f_1 и $f(h/2, w_1)$, можно рассчитать уточнённое значение $y_1 = y(h) = y_0 + \int_0^h f(t, y(t)) dt$ по формуле Симпсона, что даст ошибку порядка h^5 . Ситуация не изменится, даже если вначале значение y_1 будет задано не точно, а с ошибкой порядка h^4 .

Если эту же самую процедуру повторять на следующих шагах, то возникает аналогичная (6) система соотношений для определения нужных нам значений $y_i = y(t_i)$ и «вспомогательных» значений $w_i = y(t_i - h/2)$:

$$\begin{aligned} w_i &= \frac{y_{i-1} + y_i}{2} + \frac{f(t_{i-1}, y_{i-1}) - f(t_i, y_i)}{8} h; \\ z_i &= y_{i-1} + \frac{h}{6} [f(t_{i-1}, y_{i-1}) + 4f(t_i - h/2, w_i) + f(t_i, y_i)]; \quad (i = 1, 2, \dots, N) \\ y_i &= z_i. \end{aligned} \quad (8)$$

Значения y_1 при этом будут найдены с ошибкой порядка h^4 .

Пример 3. Будем решать то же уравнение, что и в примере 1. Точные и рассчитанные данным методом при разных шагах значения $y(t)$ представлены в следующей табл. 3.

Таблица 3. Рассчитанные и точные значения $y(t)$

t	Точные	Длительность шага h				
		0,01	0,02	0,05	0,1	0,2
0	1	1	1	1	1	1
0,2	1,1815	1,1815	1,1815	1,1815	1,1815	1,1815
0,4	1,4172	1,4172	1,4172	1,4172	1,4172	1,4172
0,6	1,7313	1,7313	1,7313	1,7313	1,7313	1,7313
0,8	2,1626	2,1626	2,1626	2,1626	2,1626	2,1626
1	2,7778	2,7778	2,7778	2,7778	2,7778	2,7777
1,2	3,6982	3,6982	3,6982	3,6982	3,6982	3,6981
1,4	5,1653	5,1653	5,1653	5,1653	5,1653	5,1651
1,6	7,7160	7,7160	7,7161	7,7160	7,7160	7,7153
1,8	12,7551	12,7551	12,7551	12,7551	12,7549	12,7517
2	25,0000	25,0000	25,0001	24,9999	24,9984	24,9744

Сравнение с табл. 1 показывает, что этот метод обеспечивает более высокую точность расчетов, правда, за счёт введения дополнительного массива, заполняемого формулами для вычисления значений переменных w_i .

Как и в первом методе, последнее равенство в (8) может быть заменено на (7), что удобно при сравнительно больших шагах h . Однако данный метод, в отличие от первого, допускает расчёты и с автоматическим выбором шага.

4. Заключение

Предложены два метода численного решения обыкновенных дифференциальных уравнений. В отличие от известных методов, в первом из них используется иное сочетание квадратурных формул, во втором – интерполяционный многочлен. Оба метода обеспечивают такой же порядок точности, что и известный метод Рунге-Кутты, но значительно проще при формировании соответствующих электронных таблиц в EXCEL.

Эти методы применимы также для решения систем обыкновенных дифференциальных уравнений, а также некоторых интегральных и интегро-дифференциальных уравнений.

Библиография:

1. Бахвалов Н.С. (1973). Численные методы (анализ, алгебра, обыкновенные дифференциальные уравнения). Москва: Наука.
2. Меркулова Н. Н., Михайлов М. Д. (2014). Методы приближенных вычислений : [учеб. пособие]. Томск: Издательский дом Томского государственного университета.
3. Мышенков В. И., Мышенков Е.В. (2005). Численные методы. Ч.2. Численное решение обыкновенных дифференциальных уравнений. Учебное пособие для студентов специальности 073000. М.: МГУЛ.
4. Пименов В. Г. (2014). Численные методы : в 2 ч. Ч. 2 : [учеб. пособие] / В. Г. Пименов, А. Б. Ложников. Екатеринбург: Изд-во Уральского федерального университета.
5. Atkinson K., Han W., Steward D. (2009). Numerical solution of ordinary differential equations. Hoboken, New Jersey: John Wiley & Sons, Inc.

Solving ordinary differential equations in EXCEL using quadrature and interpolation formulas

Sergey Smolyak

Principal Scientific Researcher, Central Economics and Mathematics Institute, Russian Academy of Sciences

Moscow, Russian Federation, Nakhimovsky prospect 47

Abstract

We propose two simple methods for numerically solving the Cauchy problem for ordinary differential equations. They provide high accuracy and are suitable for use in EXCEL and other spreadsheets.

Keywords: ordinary differential equations, numerical solution, spreadsheets

Publication date: 26.07.2021

Citation link:

Smolyak S. Solving ordinary differential equations in EXCEL using quadrature and interpolation formulas // Vestnik CEMI – 2021. – Issue 2 [Electronic resource]. URL: <https://cemi.jes.su/S265838870015151-9-1> (circulation date: 26.07.2021). DOI: 10.33276/S265838870015151-9

Код пользователя: 7821; Дата загрузки: 26.07.2021; URL - <http://ras.jes.su/cemi/s265838870015151-9-1> Все права защищены.