

ЭФФЕКТИВНЫЙ АЛГОРИТМ РЕШЕНИЯ СИСТЕМЫ ДИФФЕРЕНЦИАЛЬНЫХ УРАВНЕНИЙ ДВИЖЕНИЯ

Степаньянц В.А., Львов Д.В.

Институт прикладной математики им. М.В. Келдыша РАН
125047, Москва, Миусская площадь, 4
E-mail: stepan@kiam1.rssi.ru

Описывается метод восьмого порядка точности для решения дифференциальных уравнений движения. На каждом шаге изменения аргумента при вычислении правых частей используются четыре точки, являющиеся корнями полинома Лежандра четвертой степени, что обеспечивает максимальный восьмой порядок приближения квадратурных формул к интегралу и сводит к минимуму ошибки округления, накапливающиеся при вычислениях. В качестве начального приближения при вычислении значения приращения функции на очередном шаге используются значения, последовательно вычисленные методом Адамса, адаптированного для работы на неравномерной сетке, далее выполняется одна итерация неявного метода Рунге-Кутты. Таким образом, на каждой из четырех точек шага производится два вычисления правых частей уравнений. Коэффициенты квадратурных формул вычислены заранее для всех случаев работы алгоритма на участках постоянного шага интегрирования, а также при его уменьшении и увеличении. Это сводит к минимуму вычислительные затраты.

AN EFFECTIVE SOLVING ALGORITHM FOR DIFFERENTIAL EQUATIONS OF MOTION

Stepaniants V.A., Lvov D.V.

Keldysh Institute of Applied Mathematics Russian Academy of Sciences
4, Miusskaya Sq., Moscow, 125047

The paper presents eighth-order method for solution of differential equation. Legendre's nodes are used as argument of derivatives computing. That is why there are only four points per step for derivatives computing. This results decrease of numerical noise. Adams type predictor-corrector use to obtain initial values for single iteration of implicit Runge-Kutta algorithm, so the derivatives are computed twice for the each step. All quadrature coefficients which integrator use are calculated beforehand (for uniform step, increasing step, decreasing step) to obtain minimum possible volume of computation.

Введение

Численное интегрирование дифференциальных уравнений движения является важным средством решения задач прикладной небесной механики. В эффективной программе численного интегрирования должна быть предусмотрена возможность изменение величины шага. Для расчета движения тел по высоко эксцентричным или перелетным орбитам, то есть там, где возникает необходимость интегрирования с переменным шагом, обычно используются явные методы Рунге-Кутты. В случае многошаговых методов эта процедура вызывает некоторые затруднения. Формулы большей части существующих многошаговых методов служат для работы на равноотстоящих точках. Меньшую часть составляют методы, расчетные формулы которых приведены в соответствие узлам неравномерной сетки, например алгоритмы, предложенные в работах [1] и [2]. Однако эти и другие такого рода алгоритмы характеризуются большими накладными расходами, связанными с потерями машинного времени на вычисление коэффициентов разностных схем.

В настоящей работе предложен алгоритм численного интегрирования, позволяющий обойти эту проблему. В основу алгоритма положен синтез метода Адамса, адаптированного для не равноотстоящих узлов, и неявного метода Рунге-Кутты. В соответствии с классификацией, принятой в [3] это многошаговый девятистадийный предиктор-корректор.

Коэффициенты разностных формул вычисляются не в процессе интегрирования, а заранее, на этапе разработки метода. При этом предусмотрены все случаи расположения узловых точек при увеличении, уменьшении и сохранении шага интегрирования.

Второй особенностью алгоритма является выбор узловых точек. Узлы разбиения шага совпадают с корнями полинома Лежандра, что обеспечивает наивысшую алгебраическую степень точности на завершающей стадии вычисления значений интегрируемых функций. В результате при сохранении 8-го порядка точности для 8 вычислений правых частей мы получаем меньшее значение остаточного члена и меньшее влияние ошибок округления, так как итоговое решение зависит не от восьми, а только от четырех значений правых частей.

Вычисление коэффициентов разностных формул

Систему дифференциальных уравнений и начальные условия запишем в форме

$$\begin{aligned} \frac{d\mathbf{x}}{dt} &= \mathbf{f}(t, \mathbf{x}(t)) \\ \mathbf{x}(t_0) &= \mathbf{x}_0 \end{aligned} \quad (1)$$

где

$$\mathbf{x} = (x_1, \dots, x_n)^T, \quad \mathbf{f} = (f_1, \dots, f_n)^T.$$

Предположим, что нам известны значения правых частей \mathbf{f}_i в точках t_i , $i = 1, \dots, n$ и значение вектора интегрируемых функций $\mathbf{x}(T_0)$ в начальный момент времени. Значение $\mathbf{x}(T)$ определяется соотношением

$$\mathbf{x}(T) = \mathbf{x}(T_0) + \int_{T_0}^T \mathbf{f}(t, \mathbf{x}(t)) dt. \quad (2)$$

Пока мы не устанавливаем никаких ограничений на взаимное расположение системы точек t_i, T_0, T . Пределы интегрирования T_0, T могут располагаться как внутри, так и вне интервала $[t_1, t_n]$.

Вводя обозначения:

$$g_i = \frac{t_i - T_0}{h}, \quad i = 1, \dots, n, \quad g = \frac{T - T_0}{h}, \quad \tau = \frac{t - T_0}{h},$$

интеграл в формуле (2) можно записать в виде

$$\int_{T_0}^T \mathbf{f}(t, \mathbf{x}(t)) dt = \int_{T_0}^{T_0+gh} \mathbf{f}(T_0 + \tau h, \mathbf{x}(T_0 + \tau h)) d\tau h = h \int_0^g \mathbf{f}(T_0 + \tau h, \mathbf{x}(T_0 + \tau h)) d\tau, \quad (3)$$

где h – текущий шаг интегрирования. Набор величин g, g_0, g_1, \dots, g_n , зависящий только от соотношения интервалов между точками, будем называть шаблоном сетки.

Значение подынтегральной функции можно приблизить интерполяционным многочленом

$$\mathbf{p}(t) = \sum_{k=0}^{n-1} \mathbf{p}_k \tau^k, \quad (4)$$

принимая значения \mathbf{f}_i в точках $\tau_i = g_i$, $i = 1, \dots, n$. Коэффициенты \mathbf{p}_k можно определить из соотношения

$$\sum_{k=0}^{n-1} \mathbf{p}_k g_i^k = \mathbf{f}_i,$$

по известным конечным формулам, или численно решая систему линейных уравнений

$$\mathbf{A}\mathbf{P} = \mathbf{f}, \text{ где } \mathbf{A} = \begin{Bmatrix} 1 & g_1 & \dots & g_1^{n-1} \\ 1 & g_2 & \dots & g_2^{n-1} \\ \dots & \dots & \dots & \dots \\ 1 & g_n & \dots & g_n^{n-1} \end{Bmatrix}, \mathbf{P} = \begin{Bmatrix} \mathbf{P}_0 \\ \mathbf{P}_1 \\ \dots \\ \mathbf{P}_{n-1} \end{Bmatrix}.$$

Для нас важно, что коэффициенты многочлена $\mathbf{P}(t)$ можно представить в как линейную комбинацию правых частей

$$\mathbf{P}_k = \sum_{j=1}^n a'_{kj} \mathbf{f}_j,$$

где $a'_{kj} = a'_{kj}(n, g_1, \dots, g_n)$, $j = 1, \dots, n$, $k = 0, \dots, n-1$ — элементы матрицы \mathbf{A}^{-1} , зависящие только от шаблона сетки. Заменяя в выражении (3) подынтегральную функцию на многочлен (4) и выполняя интегрирование, получим

$$h \int_0^g \sum_{k=0}^{n-1} \mathbf{P}_k \tau^k d\tau = h \sum_{k=0}^{n-1} \frac{\mathbf{P}_k g^{k+1}}{k+1} = h \sum_{k=0}^{n-1} \left(\sum_{j=1}^n a'_{kj} \mathbf{f}_j \right) \frac{g^{k+1}}{k+1} = h \sum_{j=1}^n \mathbf{f}_j \left(\sum_{k=0}^{n-1} a'_{kj} \frac{g^{k+1}}{k+1} \right),$$

В окончательном виде выражение для приближенного решения (2) запишем в виде

$$\mathbf{x}(T) \approx \mathbf{x}(T_0) + h \sum_{j=1}^n c_j \mathbf{f}_j \quad (5)$$

где

$$c_j = \Phi(n, j, g, g_0, g_1, \dots, g_n) = \sum_{k=0}^{n-1} a'_{kj} \frac{g^{k+1}}{k+1}, \quad j = 1, \dots, n \quad (6)$$

коэффициенты, зависящие от шаблона сетки. Процедура Φ вычисления коэффициентов c_j используется как вспомогательный инструмент для получения различных вариантов разностных формул численного интегрирования. Например, набор величин $g = 1$, $g_0 = 0$, $g_1 = 0$, $g_2 = -1$, ..., $g_n = -n + 1$ дает, в результате применения этой процедуры, коэффициенты широко известной экстраполяционной формулы Адамса для равноотстоящих узлов. Значения $c_j = \Phi(n, j, 1, 0, 1, 0, \dots, -n + 2)$, $j = 1, \dots, n$ представляют собой коэффициенты интерполяционной формулы Адамса.

Схема работы алгоритма

Вернемся к уравнению (2). Если узлы разбиения отрезка $[T_0, T]$ совпадают с корнями полинома Лежандра, достигается наивысшая алгебраическая степень точности при вычислении интеграла по формуле (5). На очередном шаге интегрирования неизвестные значения \mathbf{f}_j для четырех узлов определяются двумя последовательными приближениями. В таблице 1 схематически представлена последовательность вычисления начального приближения правых частей дифференциальных уравнений в узловых точках. Знаком “плюс” обозначены точки, по которым строится аппроксимация. Звездочкой — моменты, для которых вычисляются значения интегрируемых функций и правых частей уравнений (1) h — текущий шаг интегрирования, h_{-1} и h_{-2} — два предшествующих шага, G_1, G_2, G_3, G_4 — значения корней полинома Лежандра 4-й степени на отрезке $[0, 1]$.

Таблица 1.

h_{-2}				h_{-1}				h			
$h_{-2}G_1$	$h_{-2}G_2$	$h_{-2}G_3$	$h_{-2}G_4$	$h_{-1}G_1$	$h_{-1}G_2$	$h_{-1}G_3$	$h_{-1}G_4$	hG_1	hG_2	hG_3	hG_4
		+	+	+	+	+	+	*			
			+	+	+	+	+	+	*		
				+	+	+	+	+	+	*	
					+	+	+	+	+	+	*

Очевидно, что такая схема полностью совпадает с методом Адамса для не равноотстоящих точек. Отличие состоит лишь в выборе узлов сетки.

Для построения следующего приближения значений \mathbf{f}_j используются 8 точек, принадлежащих двум последовательным шагам h и h_{-1} . В таблице 2 содержатся значения величин g, g_0, g_1, \dots, g_n использующихся для получения соответствующих коэффициентов в формуле (5). Верхняя строка указывает номер стадии. Величины $\rho_{-1} = h_{-1}/h$ и $\rho_{-2} = h_{-2}/h$ характеризуют относительное изменение двух предшествующих шагов интегрирования. Правые части уравнений (1) вычисляются дважды в каждой из четырех точек очередного шага. Сначала на стадиях 1-4, а затем на стадиях 5-8. Итерационное определение значений \mathbf{f}_j на стадиях 5-8 можно продолжить до сходимости, но мы этого не делаем, так как уже второе приближение дает достаточно точное решение.

Таблица 2

	1	2	3	4	5	6	7	8	9
g	G_1	G_2	G_3	G_4	G_1	G_2	G_3	G_4	1
g_0	0	0	0	0	0				0
g_{-1}	$\rho_{-1}(G_4 - 1)$	G_1	G_2	G_3	G_4				G_4
g_{-2}	$\rho_{-1}(G_3 - 1)$	$\rho_{-1}(G_4 - 1)$	G_1	G_2	G_3				G_3
g_{-3}	$\rho_{-1}(G_2 - 1)$	$\rho_{-1}(G_3 - 1)$	$\rho_{-1}(G_4 - 1)$	G_1	G_2				G_2
g_{-4}	$\rho_{-1}(G_1 - 1)$	$\rho_{-1}(G_2 - 1)$	$\rho_{-1}(G_3 - 1)$	$\rho_{-1}(G_4 - 1)$	G_1				G_1
g_{-5}	$\rho_{-2}(G_4 - 1) - \rho_{-1}$	$\rho_{-1}(G_1 - 1)$	$\rho_{-1}(G_2 - 1)$	$\rho_{-1}(G_3 - 1)$	$\rho_{-1}(G_4 - 1)$				0
g_{-6}	$\rho_{-2}(G_3 - 1) - \rho_{-1}$	$\rho_{-2}(G_4 - 1) - \rho_{-1}$	$\rho_{-1}(G_1 - 1)$	$\rho_{-1}(G_2 - 1)$	$\rho_{-1}(G_3 - 1)$				0
g_{-7}					$\rho_{-1}(G_2 - 1)$				0
g_{-8}					$\rho_{-1}(G_1 - 1)$				0

Для оценки точности выполнения шага аналогичным образом вычисляются пробные значения вектора интегрируемых функций $\tilde{\mathbf{x}}(T)$. При этом значения коэффициентов c_j определяются из соотношения

$$c_j = \Phi(8, j, 1, 0, G_3, G_2, G_1, \rho_{-1}(G_4 - 1), \rho_{-1}(G_3 - 1), \rho_{-1}(G_2 - 1), \rho_{-1}(G_1 - 1), \rho_{-2}(G_4 - 1) - \rho_{-1}).$$

Ошибка выполненного шага оценивается для каждой компоненты вектора $\mathbf{x}(T)$ путем ее сопоставления с значением соответствующей компоненты пробного вектора $\tilde{\mathbf{x}}(T)$. Методика оценки аналогична [1].

Особенности реализации алгоритма

Программная реализация алгоритма включает стартовую процедуру, текущее интегрирование и процедуру выхода на заданное значение аргумента (время).

Стартовая процедура (“разгон”) использует явный метод Рунге-Кутты четвертого порядка точности. Для выбора величины следующего шага h' используется текущая — Δ и прогнозируемая — $\Delta' = (h'/h)^9 \Delta$ оценки ошибки. Для каждой компоненты вектора интегрируемых функций вычисляется отношение текущей ошибки Δ_i к заданному предельному значению ошибки ε_i . Решение об увеличении или уменьшении шага принимается в соответствии таблицей 3.

Таблица 3

Уменьшение шага	Сохранение шага	Увеличение шага
$\Delta_i / \varepsilon_i > 1$ хотя бы для одного i	$\Delta_i / \varepsilon_i \leq 1$ для всех i и $\Delta_i / \varepsilon_i \geq 0.7^9$ хотя бы для одного i	$\Delta_i / \varepsilon_i < 0.7^9$ для всех i

Очередной шаг интегрирования может быть увеличен или уменьшен по отношению к предыдущему в 0.7 раза. В таблице 4 представлены 5 возможных вариантов взаимного расположения узловых точек на текущем и двух предшествующих шагах.

Таблица 4

	ρ_{-2}	ρ_{-1}	
1	1	1	Равномерный шаг
2	0.7	0.7	1-й шаг увеличения
3	0.7	1	2-й шаг увеличения
4	1/0.7	1/0.7	1-й шаг уменьшения
5	1/0.7	1	2-й шаг уменьшения

Соотношение шагов определяется значениями ρ_{-1} и ρ_{-2} . Эти величины используются для расчета аргументов g, g_0, g_1, \dots, g_n в формуле 6. Как следует из таблицы 2 для расчета коэффициентов C_j на стадиях 1, 2 используются аргументы текущего и двух предшествующих шагов, на стадиях 3-8 — текущего и одного предшествующего шага, а на стадии 9 — только текущего шага. В соответствии с этим для работы алгоритма на стадиях 1, 2 требуется пять, а на стадиях 3-8 три набора коэффициентов. Коэффициента C_j для всех вариантов такого рода вычислены по формуле (6) и используются как константы программы интегрирования.

Численный пример

В качестве примера рассматривается численное интегрирование шести уравнений невозмущенного движения точки.

$$\frac{d\mathbf{r}}{dt} = \bar{\mathbf{v}},$$

$$\frac{d\mathbf{v}}{dt} = -\mu \frac{\mathbf{r}}{r^3}.$$

Расчеты проведены для трех орбит при шести значениях допустимой ошибки. Орбиты имеют равные значения параметра $r_\pi = 6.7$ тыс. км., различные эксцентриситеты 0., 0.1, 0.5 и соответствующие значения периодов 5.457870270 тыс. сек, 6.392333767 тыс. сек, 15.43718827 тыс. сек., $\mu = 398.6004$

На рис. 1 представлена зависимость выполненных шагов от величины допустимой ошибки. Ошибка задавалась в диапазоне от 10^{-5} до 10^{-10} одинаково для координат и скоростей составляющих вектора состояния. По оси ординат отсчитываются значения допустимой ошибки в логарифмической шкале. По оси абсцисс указано число выполненных шагов. Как сказано выше, на каждом шаге интегрирования производится 8 обращений к правым частям уравнений.

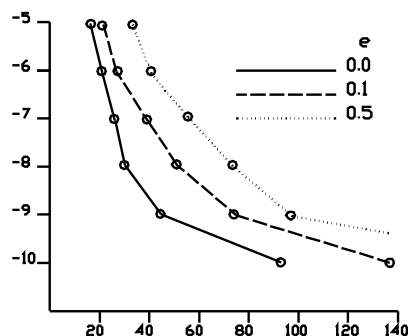


Рис.1. Зависимость предельной ошибки расчета от числа шагов.

Из приведенных графиков видно, что в диапазоне порядков от -5 до -9 при увеличении числа шагов в два раза ошибка уменьшается приблизительно на 2-3 порядка. Это приблизительно соответствует порядку точности метода ($2^9 = 512 \approx 10^{2.7}$). В диапазоне порядков -9 , -10 скорость убывания ошибки замедляется. По-видимому начинают сказываться ошибки округления.

СПИСОК ЛИТЕРАТУРЫ

1. Платонов А.К., Власова З.П., Степаньянц В.А., Многоточечный метод интегрирования с переменным шагом для обыкновенных дифференциальных уравнений. – Москва, 1976, 18с Институт прикладной математики АН СССР. Препринт №72.
2. Everhart E.. Implicit single-sequence method for integrating orbits. Celestial Mechanics 10, 1974 p.33-55.
3. Шметтер Х., Анализ методов дискретизации для обыкновенных дифференциальных уравнений. Москва, Мир, 1978, 461с.