

Алгоритм численного решения кусочно-сшитых систем

В. В. КОРОБИЦЫН, Ю. В. ФРОЛОВА
Омский государственный университет, Россия
e-mail: korobits@univer.omsk.su

В. Б. МАРЕНИЧ
Кальмарский университет, Швеция
e-mail: Valeri.Marenitch@hik.se

A PSS algorithm for numerical solving of the piecewise sewing systems is suggested. The algorithm realizes two modes of calculation: in areas of continuity and on the sewing surface. The Runge—Kutta method is used for the continuity mode. The Newton iteration method is used for the sewing mode. The developed PSS algorithm is compared with the classical methods. Usability of the PSS algorithm for piecewise sewing systems is shown.

Введение

Математические модели физических, химических, биологических и социальных систем зачастую описываются одним или несколькими дифференциальными уравнениями. Как правило, эти модели ведут себя адекватно лишь в некоторой ограниченной области фазового пространства. В природе подобные процессы имеют более сложную структуру, и поэтому приходится прибегать к использованию моделей, описываемых дифференциальными уравнениями с разрывной правой частью. Например, модель Кривана и Дейла [1] отражает поведение популяций в цепи питания, включающих всеядных особей, которые способны менять стратегию питания. А модель Мелина и Хюльгрена [2] описывает поведение резонансного преобразователя, фазовое пространство которого склеено из четырех областей.

Исследованию теории обыкновенных дифференциальных уравнений с разрывной правой частью и их применению посвящено достаточно много работ [3–12]. Среди прочих систем выделяют класс кусочно-сшитых. В частности, понятие кусочно-сшитых систем на плоскости введено в работе Н.Н. Баутина и Е.А. Леонтовича [13].

Большинство практических задач решается численными методами. Исследованию аппроксимации, устойчивости и сходимости численных методов решения обыкновенных дифференциальных уравнений посвящено много публикаций. Однако работы по численным методам решения обыкновенных дифференциальных уравнений в основном рассматривают жесткие или нежесткие непрерывные задачи либо задачи с дифференциально-алгебраическими уравнениями [14–24]. Ряд работ представляет программное обеспечение для численного исследования систем обыкновенных дифференциальных

уравнений [25–27]. Лишь немногие авторы, например, J.R. Cash, А.Н. Карп в своей статье [28], уделяют внимание построению алгоритма численного решения с переменным порядком для задач с быстроменяющейся правой частью.

В данной работе предлагается алгоритм PSS для численного решения систем обыкновенных дифференциальных уравнений, имеющих разрыв на некотором подпространстве фазового пространства. Сравняются погрешность численного решения и вычислительные затраты с традиционными численными методами решения обыкновенных дифференциальных уравнений.

1. Определение решения разрывной системы ОДУ

Рассмотрим автономную систему, описываемую набором n параметров $y = (y^1, y^2, \dots, y^n)$ в некоторой области $D \subset \mathbb{R}^n$. Пусть эта область разделена на две части D_1 и D_2 гиперповерхностью $G = \{y \in D : g(y) = 0\}$, где $g(y) : D \rightarrow \mathbb{R}$ — непрерывная функция. Таким образом, заданы области $D_1 = \{y \in D : g(y) < 0\}$ и $D_2 = \{y \in D : g(y) > 0\}$. Поведение системы описывается решением обыкновенного дифференциального уравнения в векторном виде

$$\frac{dy}{dt} = f(y), \quad y \in D, \quad (1)$$

при заданном начальном условии

$$y(t_0) = y_0, \quad y_0 \in D \setminus G. \quad (2)$$

Функция f задана следующим образом:

$$f(y) = \begin{cases} f_1(y) & \text{при } y \in D_1 \cup G, \\ f_2(y) & \text{при } y \in D_2 \cup G, \end{cases}$$

где $f_1(y)$ и $f_2(y)$ — непрерывные функции в соответствующих областях определения. Согласно данному определению функции $f(y)$ на гиперповерхности G принимают два разных значения $f_1(y^*)$ и $f_2(y^*)$, $y^* \in G$, которые в общем случае не совпадают.

Решением задачи (1)–(2) будем называть непрерывную функцию $y(t)$, производная которой почти всюду совпадает с $f(y)$. Как правило, в точке пересечения с гиперповерхностью G кривая решения непрерывна и не имеет производной. Но при стремлении к точке y^* на G слева касательная к кривой решения совпадает с $f_1(y^*)$, а справа — с $f_2(y^*)$.

Будем рассматривать системы, для которых выполняется дополнительное условие: для любой точки $y^* \in G$ векторы $f_1(y^*)$ и $f_2(y^*)$ лежат по одну сторону от касательной плоскости P к поверхности G в точке y^* . В этом случае кривая решения задачи (1)–(2) составляется из траекторий фазового пространства в D_1 и D_2 . Отсутствует так называемый скользящий режим. Предположим также, что начальная точка $y_0 \in D_1$. Тогда решение может быть представлено тремя способами:

- решение остается в области D_1 , $y = y_1(t)$ для всех $t \geq t_0$;
- решение достигает границы D за конечное время T_0 , $y = y_1(t)$ для всех $t_0 \leq t \leq T_0$;
- решение достигает гиперповерхности G за конечное время t_1 .

В третьем случае решение может пересекать гиперповерхность G конечное или бесконечное число раз. В этом случае решение можно представить в виде склеенных траекторий:

$$y(t) = y_{2i-1}(t), \quad t \in (t_{2i-2}, t_{2i-1}), \quad y_{2i-1}(t) : \mathbb{R} \rightarrow D_1,$$

$$y(t) = y_{2i}(t), \quad t \in (t_{2i-1}, t_{2i}), \quad y_{2i}(t) : \mathbb{R} \rightarrow D_2, \quad i = 1, 2, \dots,$$

$$y(t_j) = y_j^*, \quad y_j^* \in G, \quad j = 1, 2, \dots$$

Примером представленной системы могут служить кусочно-сшитые системы [13]. Одну из них приведем ниже. Она представляет интерес из-за наличия циклического решения, сшитого из обычных нециклических траекторий. Для исследования свойств численных алгоритмов подобные примеры имеют огромное значение. Аналитическое точное решение получить достаточно просто. При этом алгоритм можно исследовать на аппроксимацию к точному решению, а также на сходимость, учитывая, что решение всегда остается в некоторой ограниченной области.

2. Пример кусочно-сшитой системы с циклическим решением

Рассмотрим систему линейных обыкновенных дифференциальных уравнений, сшитую из двух систем на плоскости (y_1, y_2) вдоль прямой $y_1 = \alpha$:

$$\begin{aligned} \frac{dy_1}{dt} &= a_{11}(y_1 - c_1) + a_{12}(y_2 - d_1), \\ \frac{dy_2}{dt} &= a_{21}(y_1 - c_1) + a_{22}(y_2 - d_1), \quad (y_1, y_2) \in D_1, \end{aligned} \quad (3)$$

$$\begin{aligned} \frac{dy_1}{dt} &= a_{22}(y_1 - c_2) + a_{12}(y_2 - d_2), \\ \frac{dy_2}{dt} &= a_{21}(y_1 - c_2) + a_{11}(y_2 - d_2), \quad (y_1, y_2) \in D_2, \end{aligned} \quad (4)$$

Здесь $D_1 = \{(y_1, y_2) : y_1 < \alpha\}$, $D_2 = \{(y_1, y_2) : y_1 > \alpha\}$. В случае, когда $a_{11} = a_{22} = 0$, $a_{12} = a_{21} = 1$ решение каждой из систем

$$\begin{aligned} \frac{dy_1}{dt} &= y_2 - d_1, \\ \frac{dy_2}{dt} &= y_1 - c_1, \quad y_1 < \alpha, \end{aligned} \quad (5)$$

$$\begin{aligned} \frac{dy_1}{dt} &= y_2 - d_2, \\ \frac{dy_2}{dt} &= y_1 - c_2, \quad y_1 > \alpha, \end{aligned} \quad (6)$$

имеет одну особую точку (седло). Если $c_1 < \alpha$ и $c_2 > \alpha$, то особые точки лежат в соответствующих клетках D_1 и D_2 . Кроме того, если $d_1 = d_2$, то особые точки лежат на прямой ортогональной границе, а их сепаратрисы пересекаются в некоторых точках на этой границе. В этом случае образуется ограниченная область D_0 , заключенная между сепаратрисами. Решение внутри этой области образует цикл и не выходит за ее пределы. Цикл составлен из двух гладких кривых, склеенных на границе (рис. 1). Кривые решения описываются функциями

$$\begin{aligned} y_1(t) &= A_1 e^{t-t^*} + A_2 e^{-(t-t^*)} + A_3, \\ y_2(t) &= A_1 e^{t-t^*} + A_2 e^{-(t-t^*)} + A_4, \end{aligned} \quad (7)$$

где $A_1 = ((y_1^* - A_3) + (y_2^* - A_4))/2$, $A_2 = ((y_1^* - A_3) - (y_2^* - A_4))/2$. Величины t^* , y_1^* , y_2^* задаются начальными значениями, $A_3 = c_1$, $A_4 = d_1$ при $y_1^* < \alpha$ и $A_3 = c_2$, $A_4 = d_2$ при $y_1^* > \alpha$.

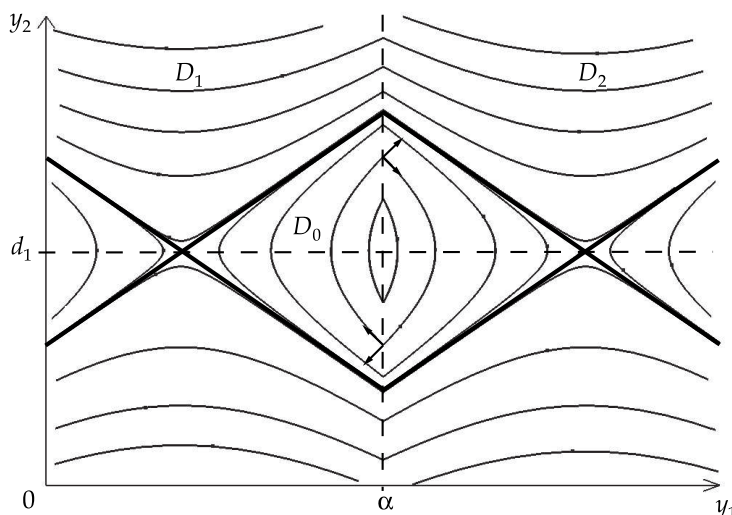


Рис. 1. Фазовый портрет системы (5)–(6)

Длина периода решения внутри D_0 вычисляется по формуле

$$T = 2 \ln \left(\frac{|A_3 - \alpha| + \sqrt{(A_3 - \alpha)^2 - 4A_1A_2}}{2A_1} \right).$$

3. Вычислительный эксперимент с традиционными методами

Исследуем поведение численного решения на задаче (5)–(6). Очевидно, что в областях D_1 и D_2 известные методы Рунге–Кутты без труда справятся с этой задачей. Но что получается при переходе решения из D_1 в D_2 и обратно? Именно на границе происходит скачок погрешности решения из-за резкого изменения правой части уравнения. И дело даже не в том, что сильно меняется правая часть, а в том, что малейшие отклонения от траектории приводят к большим отклонениям при переходе через границу G .

Исследуем погрешность численного решения для системы (5)–(6) с заданными параметрами $d_1 = 0.5$, $c_1 = 0.2$, $d_2 = 0.5$, $c_2 = 0.8$, $\alpha = 0.5$ и начальными данными $y_1(0) = 0.4999999999$, $y_2(0) = 0.3$. Приведем результаты численных экспериментов для четырех явных одношаговых методов решения систем обыкновенных дифференциальных уравнений, реализованных согласно [18]:

1) явный метод Эйлера (Explicit Euler). Одностадийная явная схема с оценкой погрешности методом Ричардсона;

2) метод Рунге–Кутты четвертого порядка (Runge–Kutta 4). Классическая четырехстадийная схема с оценкой погрешности по Ричардсону и автоматическим управлением длиной шага интегрирования;

3) метод Дорманда–Принса пятого порядка (Dormand–Prince 5(4)). Восьмистадийная схема пятого порядка с оценкой погрешности по вложенной схеме четвертого порядка;

4) метод Рунге–Кутты–Фельберга четвертого порядка (Felberg 4(5)). Наиболее распространенная вложенная схема четвертого порядка с семью стадиями, использующая встроенную схему пятого порядка для оценки локальной погрешности.

Первый эксперимент предназначен для исследования поведения схем на разрывах. Оценивалась относительная погрешность численного решения в точном решении после

преодоления одного цикла длительностью T . В этом случае точное решение попадает в начальную точку, образуя цикл. Значение численного решения показывает, насколько оно отклонилось от точного за один цикл. Все рассматриваемые методы решения имеют алгоритм оценки локальной погрешности и выбора длины шага интегрирования на основе заданной точности. Предполагается, что при уменьшении заданной точности погрешность численного решения также должна уменьшаться, обеспечивая тем самым сходимость численного метода. Однако для разрывных систем это не всегда выполняется. Кроме того, заметно значительное уменьшение порядка точности некоторых методов. Результаты эксперимента приведены на рис. 2. График зависимости изображен в логарифмическом масштабе: по горизонтальной оси откладывается порядок заданной точности ($-\log_{10}(\text{loc_tolerance})$), по вертикальной — порядок относительной погрешности решения ($-\log_{10}(\text{rel_error})$), вычисляемой по формуле

$$\text{rel_error} = \frac{\|y_n - y(T)\|}{\|y_n\|},$$

где $y(T)$ — точное решение в точке $t = T$, y_n — численное решение, полученное после n шагов, соответствующих той же точке, $\|\cdot\|$ — евклидова норма.

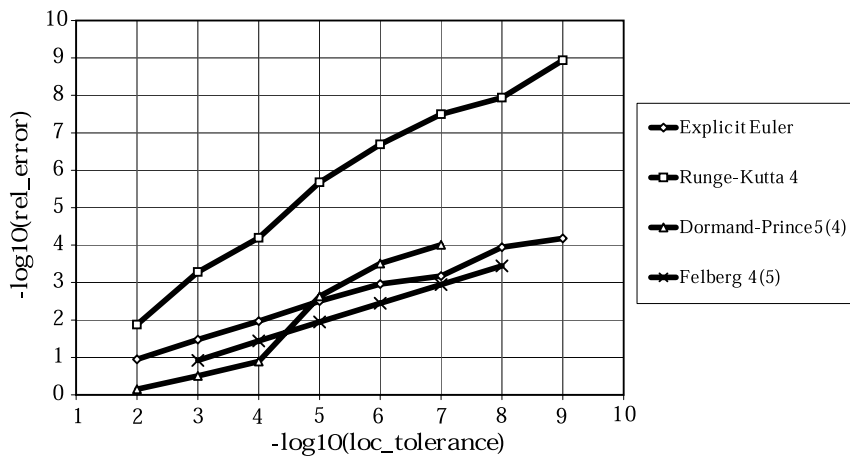


Рис. 2. Зависимость относительной погрешности решения от заданной локальной точности

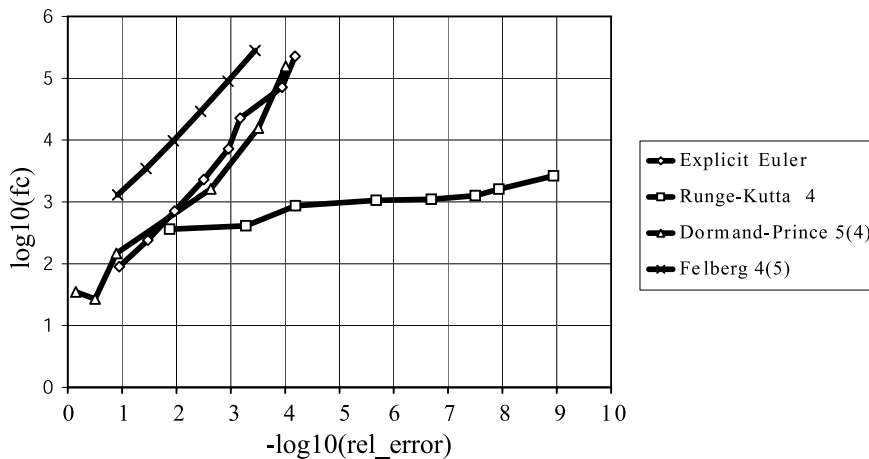


Рис. 3. Оценка вычислительных затрат для достижения заданной точности

Из диаграммы (рис. 2) видно, что метод Рунге—Кутты—Фельберга имеет бóльшую погрешность, чем явный метод Эйлера. При достаточно высокой заданной точности метод Дорманда—Принса ведет себя чуть лучше метода Эйлера. Эксперимент показал, что эти методы не достигают заданной точности. Лучшим оказался классический метод Рунге—Кутты четвертого порядка с оценкой погрешности методом Ричардсона. Он обеспечил заданную точность.

Безусловно, для нахождения численного решения требуется разный объем вычислительных затрат для различных методов. Оценим вычислительные затраты каждого метода для получения решения с относительной погрешностью, не превышающей заданную (результаты даны на рис. 3). Вычислительные затраты оцениваются количеством вычислений правых частей уравнения, которых было достаточно для обеспечения заданной точности. На рисунке логарифм от количества вычислений правых частей уравнения обозначен $(\log_{10}(fc))$. Хуже всех оказался метод Рунге—Кутты—Фельберга, почти одинаковые результаты у методов Эйлера и Дорманда—Принса. Значительно лучше работает метод Рунге—Кутты. Для задачи с разрывной правой частью механизм оценки погрешности методом Ричардсона оказался более пригодным, чем методы вложенных схем.

Хотя метод Рунге—Кутты показал себя лучше других, тем не менее используемый механизм управления длиной шага интегрирования недостаточно успешен для задач с разрывной правой частью. На рис. 4 приведена гистограмма количества шагов интегрирования в зависимости от времени на отрезке $[0, T]$ при заданной локальной точности $loc_tolerance = 10^{-8}$. Величина T — длина периода решения, в нашем примере $T = 3.2188758251282$. Решение при заданных начальных условиях начинается около границы и через половину периода $T/2 = 1.6094379125641$ достигает границы в первый раз. В этот момент происходит повышение погрешности численного решения, и механизм управления длиной шага уменьшает его, что приводит к росту числа шагов на интервале $[1.6; 1.8]$, представленного на диаграмме самым большим столбиком. Поэтому для улучшения метода интегрирования задач с разрывной правой частью необходимо уделить внимание модернизации механизма выбора шага интегрирования вблизи границы.

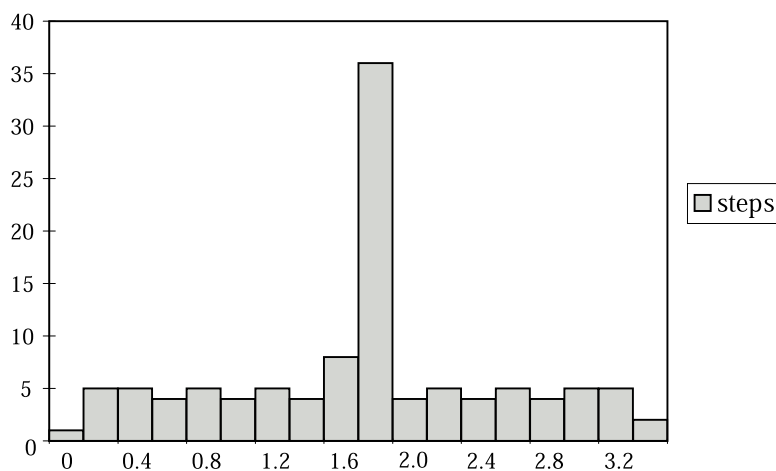


Рис. 4. Гистограмма количества шагов численного решения методом Рунге—Кутты четвертого порядка

Результаты проведенных экспериментов показали, что для преодоления разрывов в правой части системы обыкновенных дифференциальных уравнений лучше подходят методы с оценкой погрешности, основанные на экстраполяции Ричардсона, чем методы с вложенной схемой оценки погрешности. А самый удачный — классический метод Рунге—Кутты четвертого порядка. Тем не менее и этот метод нуждается в улучшении, когда речь идет о задачах с разрывной правой частью.

4. Описание алгоритма

Предлагаем алгоритм поиска решения кусочно-сшитой системы (Piecewise Sewing System) обыкновенных дифференциальных уравнений без скользящего режима.

Алгоритм PSS (поиск решения кусочно-сшитой системы). Обеспечивается поиск решения кусочно-сшитой системы обыкновенных дифференциальных уравнений на интервале времени $[a, b]$ для начального значения y_0 . Решение сохраняется в массивах T и Y . Используется схема Рунге—Кутты четвертого порядка точности для интегрирования уравнения в области непрерывности. Для поиска решения на границе используется метод экстраполяции полиномом Ньютона.

Шаг 0. (Инициализация.) Установить значения $y \leftarrow y_0$, $t \leftarrow a$, $i \leftarrow 1$. Сохранить в массив $T[0] \leftarrow t$, $Y[0] = y$. Снять все признаки $ind_bound \leftarrow false$, $ind_cham \leftarrow false$. Выбрать шаг интегрирования h .

Шаг 1. (Сделать шаг РК4.) Совершить шаг интегрирования h методом Рунге—Кутты четвертого порядка. (Получим приближенное значение решения $y_1 \approx y(t + h)$. Также будет установлен признак границы клетки $ind_bound \leftarrow true$, если при вычислении правых частей обнаружен переход в другую область.)

Шаг 2. (Решение около границы?) Если $ind_bound = true$, то перейти на шаг 3, иначе перейти на шаг 6.

Шаг 3. (Решение переходит границу?) Если $ind_cham = true$, то перейти на шаг 4, иначе перейти на шаг 5.

Шаг 4. (Поиск решения вблизи границы.) Используя таблицу последних двух принятых точек решения $T[k], Y[k]$ ($k = i - 2, i - 1$) и значений правых частей уравнения в этих точках, построить полином Ньютона $N(\theta) \approx y(t - \theta h)$. Применяя итерационный метод Ньютона, вычислить значение параметра $\theta < 0$ для достижения нужной точности приближения к границе. Определить значения полинома до границы $y_1 = N(\theta_1)$ и за границей $y_2 = N(\theta_2)$, выбирая подходящие θ_1, θ_2 . Сохранить результаты $T[i] \leftarrow t - \theta_1 h$, $Y[i] \leftarrow y_1$, $T[i + 1] \leftarrow t - \theta_2 h$, $Y[i + 1] \leftarrow y_2$, $i \leftarrow i + 2$ и увеличить время $t \leftarrow t - \theta_2 h$. Снять признаки $ind_cham \leftarrow false$, $ind_bound \leftarrow false$. Перейти на шаг 7.

Шаг 5. (Определение оптимального шага до границы.) Вычислить длину шага интегрирования h , обеспечивающую наибольшее приближение к границе области. Установить признак перехода в другую область $ind_cham \leftarrow true$. Перейти на шаг 7.

Шаг 6. (Сделать два шага РК4.) Совершив два шага интегрирования $h/2$ методом Рунге—Кутты четвертого порядка, получить y_4 . Оценить погрешность решения и выбрать новый шаг интегрирования. Если погрешность в пределах заданной точности, то сохранить результат $T[i] \leftarrow t + h$, $Y[i] \leftarrow y_4$, $i \leftarrow i + 1$ и увеличить время $t \leftarrow t + h$.

Шаг 7. (Закончить вычисления?) Если $t \leq b$, то вернуться к шагу 1, иначе закончить вычисления.

В основе алгоритма выступает классический метод Рунге—Кутты четвертого порядка. Различие проявляется на линии сшивания областей, там где правая часть урав-

нения терпит разрыв. При приближении к границе происходит переключение режима *ind_bound*. Приближение определяется попаданием в другую область при вычислении правых частей в схеме Рунге—Кутты. Если хотя бы для одной функции потребовалось значение из другой области, то считаем, что решение приблизилось к границе.

В режиме *ind_bound* (шаг 5) определяется шаг интегрирования, который обеспечивает приближение к границе, при этом остается в той же области. Шаг вычисляется по формуле

$$h_{new} = \alpha \frac{g(y_i)}{-\nabla g(y_i) \cdot f(y_i, t_i)},$$

где y_i — принятое значение решения в момент времени t_i , $g(y)$ — функция границы, $-\nabla g(y)$ — градиент к границе, $f(y, t)$ — значение правой части в точке y в момент времени t , α — коэффициент, определяемый опытным путем. В знаменателе дроби стоит скалярное произведение градиента к границе и вектора правых частей системы, описывающего направление касательной к кривой решения. Таким образом, вся дробь оценивает время, за которое решение преодолет расстояние до границы. Для того чтобы наверняка не пересечь границу, выбираем коэффициент $\alpha = 0.9$.

При численном интегрировании методом Рунге—Кутты требуется вычисление правых частей в различных точках фазового пространства и, если одна из точек попадает в другую область, получаемое решение имеет большую погрешность. Эту погрешность приходится компенсировать значительным уменьшением шага интегрирования, как это происходит при традиционном методе Рунге—Кутты. Идея улучшения алгоритма состоит в том, что мы будем использовать найденные точки решения для прогноза поведения кривой решения при приближении к границе, не прибегая к значениям правой части в другой области (шаг 4). Воспользуемся интерполяционной формулой Ньютона [29] для интерполирования назад:

$$N(\theta) = y(t_i - \theta h) = y_i + (-\theta h) \left(f_i + (-\theta)(f_i - d + (1 - \theta)(f_i - 2d + f_{i-1})) \right),$$

где используются два предыдущих шага $y_{i-1} \approx y(t_i - h)$, $y_i \approx y(t_i)$ и значения правых частей в этих точках f_{i-1} , f_i ; значение $d = (f_i - f_{i-1})/h$; параметр θ определяет промежуточные значения. Но мы применяем эту формулу не для интерполяции, а для экстраполяции с отрицательными значениями параметра θ . Тогда получим полином Ньютона, приближающий кривую решения не только внутри интервала $[t_i - h, t_i]$, но и в интервале $[t_i, t_i + h]$. Именно в этом интервале найдется пересечение кривой решения с поверхностью границы. Мы воспользовались формулой интерполирования назад, поскольку она обеспечивает более высокую точность для экстраполирования. Как показали эксперименты, увеличение степени полинома Ньютона не дают выигрыша в точности или скорости сходимости. Построенный полином используется для нахождения пересечения кривой решения с поверхностью границы. Для этого применяем итерационный метод Ньютона с формулой изменения параметра θ :

$$\theta \leftarrow \theta - \beta \frac{g(N(\theta))}{-\nabla g(y_i) f(y_i, t_i) h}.$$

В этой формуле используем фиксированное значение для скорости изменения кривой решения (знаменатель дроби). Это значение мы определяем в точке y_2 , такое приближение оказывается вполне допустимым. Коэффициент β выбирается экспериментально

от 0.5 до 0.9 для обеспечения наибольшей скорости сходимости. Условием завершения итерационной процедуры является выполнение выражения

$$\Delta\theta \leq \kappa \frac{Tol}{\eta h},$$

где κ — коэффициент от 0.01 до 0.1, Tol — заданная точность, $\eta = \tau/(1-\tau)$, $\tau = \Delta\theta/\Delta\theta_2$. Величина $\Delta\theta$ определяет изменение параметра θ , $\Delta\theta_2$ — значение $\Delta\theta$ на предыдущем шаге итерационной процедуры. Как показала практика, число итераций зависит от заданной точности и находится в пределах от 5 до 30 итераций для точности от 10^{-3} до 10^{-10} .

5. Вычислительный эксперимент с алгоритмом PSS

Результаты проведенного эксперимента показаны на рис. 5. На диаграммах продемонстрирована зависимость относительной погрешности численного решения от заданной локальной точности и зависимость вычислительных затрат от погрешности решения. Из рисунка видно, что погрешности обоих методов (классический метод — Runge—Kutta 4, предложенный алгоритм — Algorithm PSS) одинаковы и соответствуют задан-

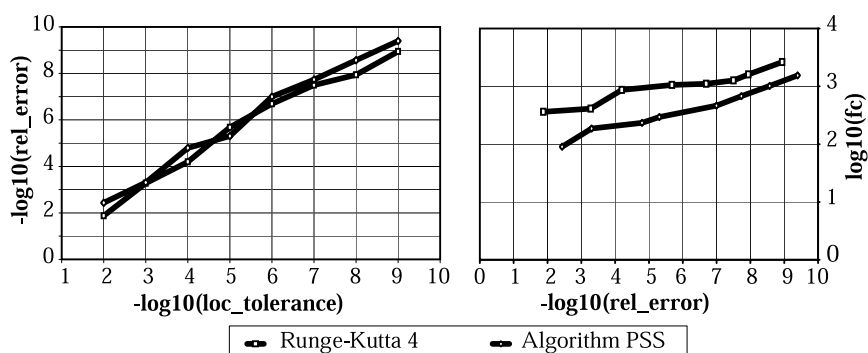


Рис. 5. Зависимость относительной погрешности решения и вычислительных затрат от заданной локальной точности для классического метода и алгоритма PSS

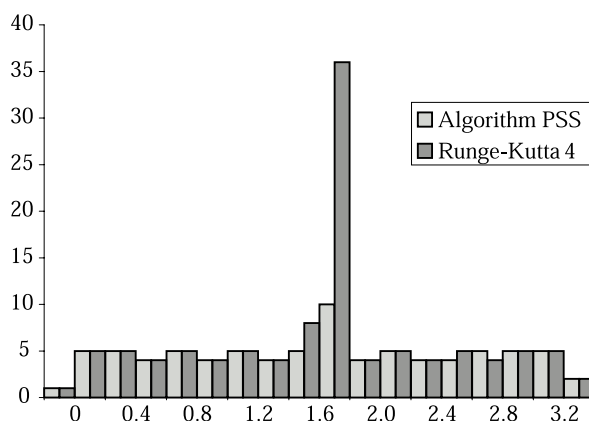


Рис. 6. Гистограмма сравнения количества шагов численного решения методом Рунге—Кутты и алгоритма PSS

Оценка времени решения для метода Рунге—Кутты и алгоритма PSS

<i>Tol</i>	Runge—Kutta 4		Algorithm PSS	
	100 циклов	1000 циклов	100 циклов	1000 циклов
10^{-4}	13	126	3	26
10^{-5}	17	136	5	42
10^{-6}	22	215	6	62
10^{-7}	28	268	10	98
10^{-8}	33	343	14	153
10^{-9}	44	443	23	241

ной локальной точности. Однако вычислительные затраты на выполнение алгоритмов разные. И они существенно меньше для предложенного алгоритма.

Снижение уровня вычислительных затрат происходит благодаря вводу дополнительного механизма управления длиной шага при переходе через границу между областями. Резкое уменьшение числа шагов в районе границы продемонстрировано на гистограмме (рис. 6).

В предложенном алгоритме помимо самой схемы Рунге—Кутты требуется составлять полином Ньютона и находить точку пересечения этого полинома с поверхностью границы. Возникает вопрос: время, потраченное на дополнительные вычисления по поиску пересечения кривой решения с границей итерационным методом Ньютона, больше, чем время, потраченное на дополнительные шаги по методу Рунге—Кутты, или нет? Однако эксперименты показывают, что время вычисления полинома значительно меньше времени просчета мелких шагов около границы методом Рунге—Кутты с выбором шага по оценке Ричардсона (результаты экспериментов даны в таблице). Эксперименты проведены для задачи (5)–(6) в одной программе с разными алгоритмами решения. Но поскольку время на поиск решения на отрезке одного периода слишком мало, то для оценки использовалась задача на интервале 100 и 1000 периодов при разной заданной локальной точности *Tol*. Время дано в секундах при расчетах на компьютере с процессором Intel Pentium-M 1.8 ГГц с оперативной памятью 256 Мбайт.

Результаты экспериментов показывают целесообразность использования предложенного алгоритма PSS для решения кусочно-сшитых систем дифференциальных уравнений с разрывной правой частью.

Список литературы

- [1] KŘIVAN V., DIENL S. Adaptive omnivory and species coexistence in tri-trophic food webs // *Theoretical Population Biology*. 2005. N 67. P. 85–99.
- [2] MELIN J., HULTGREN A. A limit cycle of a resonant converter // Engell H.G.S., Zaytoon J. *Analysis and design of hybrid systems 2003*. IFAC, Elsevier, 2003. P. 169–174.
- [3] АЙЗЕРМАН М.А., ПЯТНИЦКИЙ Е.С. Основы теории разрывных систем I, II // *Автоматика и телемеханика*. 1974. № 7. С. 33–47; № 8. С. 39–61.
- [4] ДЕРР В.Я., КИНЗЕВУЛАТОВ Д.М. Дифференциальные уравнения с обобщенными функциями, допускающими умножение на разрывные функции // *Вестн. Удм. ун-та. Ижевск*. 2005. № 1. С. 35–58.
- [5] ЗУЕВ А.В. О периодических решениях обыкновенных дифференциальных уравнений с разрывной правой частью // *Математические заметки*. 2006. Т. 79, № 4. С. 560–570.

- [6] ЗУЕВ А.В., ФИЛИППОВ В.В. О задаче Неймана для обыкновенного дифференциального уравнения с разрывной правой частью // Дифференциальные уравнения. 2005. Т. 41, № 6. С. 755–760.
- [7] МАТРОСОВ И.В. О единственности справа решений невырожденных алгебро-дифференциальных уравнений с разрывами // Автоматика и телемеханика. 2007. № 1. С. 11–19.
- [8] САМОЙЛЕНКО А.М., ПЕРЕСТЮК Н.А. Дифференциальные уравнения с импульсным воздействием. Киев: Вища школа, 1987. 288 с.
- [9] СЕСЕКИН А.Н. О нелинейных дифференциальных уравнениях в классах функций ограниченной вариации // Дифференциальные уравнения. 1989. Т. 25, № 11. С. 1925–1932.
- [10] УТКИН В.И. Скользящие режимы в задачах оптимизации и управления. М.: Наука, 1981. 368 с.
- [11] ФИЛИППОВ А.Ф. Дифференциальные уравнения с разрывной правой частью. М.: Наука, 1985. 224 с.
- [12] SARRICO C.O.R. The linear Cauchy problem for a class of differential equations with distributional coefficients // Portugalie Math. 1995. Vol. 52. P. 379–390.
- [13] БАУТИН Н.Н., ЛЕОНТОВИЧ Е.А. Методы и приемы качественного исследования динамических систем на плоскости. М.: Наука, 1990. 486 с.
- [14] АЙДА-ЗАДЕ К.Р. О численном решении систем дифференциальных уравнений с нелокальными условиями // Вычисл. технологии. 2004. Т. 9, № 1. С. 11–25.
- [15] ДЕККЕР К., ВЕРВЕР Я. Устойчивость методов Рунге—Кутты для жестких нелинейных дифференциальных уравнений. М.: Мир, 1988. 334 с.
- [16] НОВИКОВ Е.А. Неоднородный метод второго порядка для жестких систем // Вычисл. технологии. 2005. Т. 10, № 2. С. 74–86.
- [17] ОЛЕМСКОЙ И.В. Явный метод типа Рунге—Кутты пятого порядка // Вычисл. технологии. 2005. Т. 10, № 2. С. 87–105.
- [18] ХАЙРЕР Э., НЁРСЕТТ С., ВАННЕР Г. Решение обыкновенных дифференциальных уравнений. Нежесткие задачи. М.: Мир, 1990. 512 с.
- [19] ХАЙРЕР Э., ВАННЕР Г. Решение обыкновенных дифференциальных уравнений. Жесткие и дифференциально-алгебраические задачи. М.: Мир, 1999. 685 с.
- [20] DEUFLHARD P., HAIRER E., ZUGCK J. One-step and extrapolation methods for differential-algebraic systems // Numerische Mathematik. 1987. Vol. 51, N 5. P. 501–516.
- [21] GUPTA G.K., SACKS-DAVIS R., TESCHER P.E. A review of recent developments in solving ODEs // ACM Computing Surveys. 1985. Vol. 17, N 1. P. 5–47.
- [22] HAIRER E., LUBICH CH., ROCHE M. Error of Runge—Kutta methods for stiff problems studied via differential algebraic equations // BIT Numerical Mathematics. 1988. Vol. 28, N 3. P. 678–700.
- [23] NEDIALKOV N.S., PRYCE J.D. Solving differential-algebraic equations by Taylor series (II): Computing the System Jacobian // BIT Numerical Mathematics. 2007. Vol. 47, N 1. P. 121–135.
- [24] PARK T., BARTON P.I. State event location in differential-algebraic models // ACM Transactions on Modeling and Computer Simulation. 1996. Vol. 6, N 2. P. 137–165.
- [25] BERZINS M., BRANKIN R.W., GLADWELL I. The stiff integrators in the NAG library // ACM SIGNUM Newsletter. 1988. Vol. 23, N 2. P. 16–23.

- [26] ENRIGHT W.H., PRYCE J.D. Two FORTRAN packages for assessing initial value methods // ACM Transactions on Mathematical Software. 1987. Vol. 13, N 1. P. 1–27.
- [27] KAMEL M.S., ENRIGHT W.H., MA K.S. ODEXPERT: an expert system to select numerical solvers for initial value ODE systems // ACM Transactions on Mathematical Software. 1993. Vol. 19, N 1. P. 44–62.
- [28] CASH J.R., KARP A.H. A variable order Runge–Kutta method for initial value problems with rapidly varying right-hand sides // ACM Transactions on Mathematical Software. 1990. Vol. 16, N 3. P. 201–222.
- [29] ПЕТРОВСКАЯ Н.Б. Применение метода Ньютона в расчетных схемах высокого порядка для стационарных задач // Вычисл. технологии. 2005. Т. 10, № 2. С. 106–113.

Поступила в редакцию 29 октября 2007 г.