

КОНТРОЛЬ УСТОЙЧИВОСТИ МЕТОДА ФЕЛЬБЕРГА СЕДЬМОГО ПОРЯДКА ТОЧНОСТИ*

Е. А. НОВИКОВ

Институт вычислительного моделирования СО РАН

Красноярск, Россия

e-mail: novikov@icm.krasn.ru

Ю. В. ШОРНИКОВ

Новосибирский государственный технический университет, Россия

An inequality for the stability control of Felberg method of 7th order of accuracy is formulated, which is based on the evaluation of maximum eigenvalue of Jacoby matrix by the power-law method. This evaluation does not increase the number of calculations of the right-hand side of ODE. The results of computations are provided, which confirm the almost double increase of efficiency of the method due to the additional stability control.

Введение

В ряде случаев возникает необходимость численного решения задачи Коши для жестких систем обыкновенных дифференциальных уравнений алгоритмами на основе явных методов. Это требуется, например, при большой размерности системы с громоздкой правой частью. Алгоритмы интегрирования, основанные на неявных или полуявных формулах, как правило, используют обращение матрицы Якоби, что в данном случае есть отдельная трудоемкая задача. Кроме того, получение элементов матрицы Якоби и составление подпрограммы ее нахождения требуют от вычислителя больших затрат личного времени. В такой ситуации предпочтительнее применять алгоритмы, основанные на явных формулах, если жесткость задачи позволяет за разумное время получить приближение к решению. Современные алгоритмы на основе явных методов в большинстве своем не приспособлены для решения жестких задач по следующей причине. Обычно алгоритм управления шагом интегрирования строится на контроле точности численной схемы. Это естественно, потому что основным критерием является точность нахождения решения. Однако при использовании алгоритмов интегрирования на основе явных формул для решения жестких задач этот подход приводит к потере эффективности и надежности, потому что на участке установления вследствие противоречивости требований точности и устойчивости шаг интегрирования раскачивается, а также к большому количеству возвратов (повторных вычислений решения), а шаг выбирается значительно меньше максимально допустимого.

*Работа выполнена при финансовой поддержке Российского фонда фундаментальных исследований (грант № 05-01-00579-а).

© Институт вычислительных технологий Сибирского отделения Российской академии наук, 2006.

Этого можно избежать, если наряду с точностью контролировать и устойчивость численной схемы.

В настоящее время можно выделить два подхода к контролю устойчивости [1, 2]. Первый связан с оценкой максимального собственного числа матрицы Якоби через ее норму с последующим контролем (наряду с точностью) неравенства $h\|f_y\| \leq D$ [1], где положительная постоянная D связана с размером области устойчивости. Ясно, что для явных методов, где матрица Якоби не участвует в вычислительном процессе, это приводит дополнительно к ее нахождению и, следовательно, к значительному увеличению вычислительных затрат. Второй подход основан на оценке максимального собственного числа матрицы Якоби степенным методом через приращения правой части системы дифференциальных уравнений с последующим контролем неравенства $h|\lambda_{\max}| \leq D$ [2]. Во всех рассмотренных авторами ситуациях такая оценка фактически не приводит к увеличению вычислительных затрат [3].

В настоящей статье с применением предложенного в [2] способа оценки максимального собственного числа матрицы Якоби построено неравенство для контроля устойчивости метода Фельберга седьмого порядка точности. Приведены результаты расчетов, подтверждающие повышение эффективности алгоритма интегрирования почти в два раза за счет контроля устойчивости.

1. Метод Фельберга

Рассмотрим задачу Коши для системы обыкновенных дифференциальных уравнений

$$y' = f(t, y), \quad y(t_0) = y_0, \quad t_0 \leq t \leq t_k, \quad (1)$$

где y и f — N -мерные вещественные вектор-функции; t — независимая переменная. Для решения (1) будем использовать явные формулы типа Рунге — Кутты следующего вида:

$$y_{n+1} = y_n + \sum_{i=1}^{13} p_{mi} k_i, \quad k_i = hf \left(t_n + \alpha_i h, y_n + \sum_{j=1}^{i-1} \beta_{ij} k_j \right), \quad (2)$$

где h — шаг интегрирования,

$$\begin{aligned} \alpha_1 &= 0, \quad \alpha_2 = \frac{2}{27}, \quad \alpha_3 = \frac{1}{9}, \quad \alpha_4 = \frac{1}{6}, \quad \alpha_5 = \frac{5}{12}, \quad \alpha_6 = \frac{1}{2}, \quad \alpha_7 = \frac{5}{6}, \\ \alpha_8 &= \frac{1}{6}, \quad \alpha_9 = \frac{2}{3}, \quad \alpha_{10} = \frac{1}{3}, \quad \alpha_{11} = 1, \quad \alpha_{12} = 0, \quad \alpha_{13} = 1, \\ \beta_{21} &= \frac{2}{27}, \quad \beta_{31} = \frac{1}{36}, \quad \beta_{32} = \frac{1}{12}, \quad \beta_{41} = \frac{1}{24}, \quad \beta_{42} = 0, \quad \beta_{43} = \frac{1}{8}, \\ \beta_{51} &= \frac{5}{12}, \quad \beta_{52} = 0, \quad \beta_{53} = -\frac{25}{16}, \quad \beta_{54} = \frac{25}{16}, \quad \beta_{61} = \frac{1}{20}, \quad \beta_{62} = \beta_{63} = 0, \\ \beta_{64} &= \frac{1}{4}, \quad \beta_{65} = \frac{1}{5}, \quad \beta_{71} = -\frac{25}{108}, \quad \beta_{72} = \beta_{73} = 0, \quad \beta_{74} = \frac{125}{108}, \quad \beta_{75} = -\frac{65}{27}, \\ \beta_{76} &= \frac{125}{54}, \quad \beta_{81} = \frac{31}{300}, \quad \beta_{82} = \beta_{83} = \beta_{84} = 0, \quad \beta_{85} = \frac{61}{225}, \quad \beta_{86} = -\frac{2}{9}, \\ \beta_{87} &= \frac{13}{900}, \quad \beta_{91} = 2, \quad \beta_{92} = \beta_{93} = 0, \quad \beta_{94} = \frac{23}{108}, \quad \beta_{95} = \frac{704}{45}, \quad \beta_{96} = -\frac{107}{9}, \\ \beta_{97} &= \frac{67}{90}, \quad \beta_{98} = 3, \quad \beta_{10,1} = -\frac{91}{108}, \quad \beta_{10,2} = \beta_{10,3} = 0, \quad \beta_{10,4} = \frac{23}{108}, \end{aligned}$$

$$\begin{aligned}
\beta_{10,5} &= -\frac{976}{135}, \beta_{10,6} = \frac{311}{54}, \beta_{10,7} = -\frac{19}{60}, \beta_{10,8} = \frac{17}{6}, \beta_{10,9} = -\frac{1}{12}, \\
\beta_{11,1} &= \frac{2383}{4100}, \beta_{11,2} = \beta_{11,3} = 0, \beta_{11,4} = -\frac{341}{164}, \beta_{11,5} = \frac{4496}{1025}, \beta_{11,6} = -\frac{301}{82}, \\
\beta_{11,7} &= \frac{2133}{4100}, \beta_{11,8} = \frac{45}{82}, \beta_{11,9} = \frac{45}{164}, \beta_{11,10} = \frac{18}{41}, \\
\beta_{12,1} &= \frac{3}{205}, \beta_{12,2} = \beta_{12,3} = \beta_{12,4} = \beta_{12,5} = 0, \beta_{12,6} = -\frac{6}{41}, \beta_{12,7} = -\frac{3}{205}, \\
\beta_{12,8} &= -\frac{3}{41}, \beta_{12,9} = \frac{3}{41}, \beta_{12,10} = \frac{6}{41}, \beta_{12,11} = 0, \beta_{13,1} = -\frac{1777}{4100}, \\
\beta_{13,2} &= \beta_{13,3} = 0, \beta_{13,4} = -\frac{341}{164}, \beta_{13,5} = \frac{4496}{1025}, \beta_{13,6} = -\frac{289}{82}, \\
\beta_{13,7} &= -\frac{2193}{4100}, \beta_{13,8} = \frac{51}{82}, \beta_{13,9} = \frac{33}{164}, \beta_{13,10} = \frac{12}{41}, \beta_{13,11} = 0, \beta_{13,12} = 1.
\end{aligned} \tag{3}$$

При значениях коэффициентов

$$\begin{aligned}
p_{71} &= \frac{41}{840}, p_{72} = p_{73} = p_{74} = p_{75} = 0, p_{76} = \frac{34}{105}, p_{77} = p_{78} = \frac{9}{35}, \\
p_{79} &= p_{7,10} = \frac{9}{280}, p_{7,11} = \frac{41}{840}, p_{7,12} = p_{7,13} = 0
\end{aligned} \tag{4}$$

схема (2), (3) имеет седьмой порядок точности [4].

2. Контроль точности и устойчивости

Согласно [4] численная формула (2), (3) с коэффициентами

$$\begin{aligned}
p_{81} &= p_{82} = p_{83} = p_{84} = p_{85} = 0, p_{86} = \frac{34}{105}, p_{87} = p_{88} = \frac{9}{35}, \\
p_{89} &= p_{8,10} = \frac{9}{280}, p_{8,11} = 0, p_{8,12} = p_{8,13} = \frac{41}{840}
\end{aligned} \tag{5}$$

имеет восьмой порядок точности. Тогда локальную ошибку δ_n метода седьмого порядка можно оценить по формуле

$$\delta_n = \sum_{i=1}^{13} (p_{8i} - p_{7i}) k_i. \tag{6}$$

В результате для контроля точности вычислений применяется неравенство

$$\|\delta_n\| \leq \varepsilon, \tag{7}$$

где $\|\cdot\|$ — некоторая норма в R^N ; ε — требуемая точность расчетов. Учитывая, что $\delta_n = O(h^8)$, шаг h^{ac} по точности выбирается по формуле

$$h^{ac} = qh, \tag{8}$$

где q находится из уравнения

$$q^8 \|\delta_n\| = \varepsilon.$$

Если $q < 1$, то происходит повторное вычисление решения (возврат) с шагом h , равным qh . В противном случае вычисляется приближенное решение, а прогнозируемый шаг находится по формуле (8). Неравенство (7) хорошо зарекомендовало себя при решении многих практических задач, оно используется и в данной работе. В дальнейшем алгоритм переменного шага на основе схемы (2) с коэффициентами (3)–(5) и неравенством для контроля точности (7) будем называть FEL78.

Чтобы построить неравенство для контроля устойчивости, применим (2) для решения линейной задачи $y' = Ay$ с постоянной матрицей A . Первые три стадии k_1 , k_2 и k_3 схемы (2) применительно к данной задаче имеют вид

$$k_1 = Xy_n, \quad k_2 = \left(X + \frac{2}{27}X^2\right)y_n, \quad k_3 = \left(X + \frac{1}{9}X^2 + \frac{1}{162}X^3\right)y_n,$$

где $X = hA$. Нетрудно видеть, что имеют место соотношения

$$12k_3 - 18k_2 + 6k_1 = \frac{2}{27}X^3y_n, \quad k_2 - k_1 = \frac{2}{27}X^2y_n.$$

Теперь оценку максимального собственного числа матрицы Якоби можно вычислить степенным методом. Введем обозначение

$$v_n = \max_{1 \leq j \leq N} \frac{|(12k_3 - 18k_2 + 6k_1)_j|}{|(k_2 - k_1)_j|}. \quad (9)$$

Тогда согласно [3] для контроля устойчивости метода Фельберга можно применять неравенство

$$v_n \leq D, \quad (10)$$

где постоянная D ограничивает интервал устойчивости. Устойчивость методов типа Рунге — Кутты обычно исследуется на скалярном тестовом уравнении

$$y' = \lambda y, \quad \operatorname{Re}(\lambda) < 0. \quad (11)$$

Применяя (2) с коэффициентами (4) для решения (11), получим, что функция устойчивости $Q_7(x)$ метода седьмого порядка точности имеет вид

$$Q_7(x) = 1 + \sum_{i=1}^{11} c_{7,i}x^i, \quad x = h\lambda,$$

где

$$\begin{aligned} c_{7,1} &= 1, & c_{7,2} &= 0.5, & c_{7,3} &= 0.166666666666667, \\ c_{7,4} &= 0.416666666666667 \cdot 10^{-1}, & c_{7,5} &= 0.833333333333333 \cdot 10^{-2}, \\ c_{7,6} &= 0.138888888888889 \cdot 10^{-2}, & c_{7,7} &= 0.19841269841270 \cdot 10^{-3}, \\ c_{7,8} &= 0.23165371472663 \cdot 10^{-4}, & c_{7,9} &= 0.23671439526314 \cdot 10^{-5}, \\ c_{7,10} &= 0.51829448771964 \cdot 10^{-7}, & c_{7,11} &= -0.43191207309970 \cdot 10^{-7}. \end{aligned}$$

С использованием [3] нетрудно убедиться, что интервал устойчивости метода седьмого порядка приблизительно равен 5. Функция устойчивости $Q_8(x)$ метода восьмого порядка точности имеет вид

$$Q_8(x) = 1 + \sum_{i=1}^{12} c_{8,i}x^i, \quad x = h\lambda,$$

где

$$\begin{aligned} c_{8,1} &= 1, & c_{8,2} &= 0.5, & c_{8,3} &= 0.16666666666667, \\ c_{8,4} &= 0.41666666666667 \cdot 10^{-1}, & c_{8,5} &= 0.83333333333333 \cdot 10^{-2}, \\ c_{8,6} &= 0.13888888888889 \cdot 10^{-2}, & c_{8,7} &= 0.19841269841270 \cdot 10^{-3}, \\ c_{8,8} &= 0.24801587301587 \cdot 10^{-4}, & c_{8,9} &= 0.23490700935724 \cdot 10^{-5}, \\ c_{8,10} &= 0.23620053064283 \cdot 10^{-6}, & c_{8,11} &= -0.25914724385982 \cdot 10^{-7}, \\ c_{8,12} &= -0.14397069103323 \cdot 10^{-7}. \end{aligned}$$

Интервал устойчивости метода восьмого порядка также приблизительно равен 5. Поэтому в неравенстве (10) положим $D = 5$. Учитывая, что $v_n = O(h)$, шаг h^{st} по устойчивости можно выбирать по формуле $h^{st} = rh$, где r вычисляется из равенства $rv_n = D$.

Оценка (9) является грубой, потому что: 1) вовсе не обязательно, чтобы максимальное собственное число было сильно отделено от остальных; 2) в степенном методе применяется мало итераций; 3) дополнительные искажения вносит нелинейность задачи (1). Поэтому здесь контроль устойчивости используется как ограничитель на размер шага интегрирования. В результате прогнозируемый шаг вычисляется по формуле

$$h := \max[h, \min(h^{ac}, h^{st})].$$

В дальнейшем алгоритм переменного шага с дополнительным контролем устойчивости численной схемы будем называть FEL78ST.

3. Компьютерное моделирование

Новые алгоритмы анализа сложных динамических систем оказываются более эффективными для прикладного специалиста, если они окружены специализированными инструментальными средствами [6]. Индустрия создания таких средств моделирования становится сама по себе фундаментальной задачей исследования. Сейчас масштаб и объем трудностей при создании инструментальных средств настолько выросли, что можно говорить, что задача их преодоления сама стала задачей науки и представляет собой проблему фундаментального значения. [7].

Программная реализация алгоритмов с контролем точности и устойчивости встроена в библиотеку системы инструментальных средств машинного анализа ИСМА [8]. Класс исследуемых объектов в ИСМА ограничен системами обыкновенных дифференциальных уравнений с запаздыванием

$$\dot{x} = f(x(t), x(t - \theta), t), \quad t > 0, \quad x(0) = x_0,$$

где $x \in R^N$ — вектор состояния; $x(t) = \psi(t)$ при $t \in [-\theta, 0)$; t — независимая переменная; $\psi(t)$ — m -мерная вектор-функция запаздывания, $m \leq N$; $\theta = (\tau_1, \dots, \tau_m)^T$ — вектор чистых запаздываний; f — нелинейная вектор-функция, удовлетворяющая условию Липшица; $x_0 = (x_{01}, \dots, x_{0N})^T$ — вектор начальных условий.

Инструментальная среда ИСМА реализована с учетом простоты описания математических моделей и ориентирована на предметного пользователя. Переход от математической модели к ее компьютерному описанию происходит переписыванием правой части во встроеном редакторе ИСМА или в любом другом процессоре, допускающем текстовый

формат данных. Языковой процессор ИСМА интерпретирует введенные данные и придает им соответствующие типы, спецификации и т. д. В результате трудоемкость перехода к компьютерной модели существенно снижается. Графический интерпретатор GRIN предоставляет предметному пользователю широкомасштабную манипуляцию графическими данными (трассировку точек, мультиоконный режим отображения, конкатенацию графических объектов, редактирование текстовых полей, импорт данных и т. д.).

По сравнению с известными отечественными (MVS, AnyLogic) и зарубежными (Simulink, Modelica) системами моделирования [9] ИСМА отличается простотой и естественной формой представления обозначенного класса моделей и не требует от предметного пользователя дополнительных знаний в области программирования, а также современных парадигм (UML, ООП). Система ИСМА имеет разнообразную библиотеку традиционных и оригинальных численных методов анализа жестких и нежестких систем и поэтому в списке современных инструментов визуального моделирования предоставляет пользователю свои функциональные преимущества.

4. Результаты расчетов

Расчеты проводились на IBM PC Athlon(tm)XP 2000 с двойной точностью в среде ИСМА. В конкретных расчетах левая часть неравенства (7) вычислялась по формуле

$$\|\delta_n\| = \max_{1 \leq j \leq N} \frac{|\delta_n^j|}{|y_n^j| + r},$$

где j — номер компоненты, r — положительный параметр. Если по j -й компоненте решения выполняется неравенство $|y_n^j| < r$, то контролируется абсолютная ошибка εr , в противном случае — относительная ошибка ε . В расчетах параметр r выбирался таким образом, чтобы по всем компонентам решения фактическая точность была не хуже задаваемой.

Ниже через isa и iwo обозначены соответственно суммарное число шагов интегрирования и число повторных вычислений решения (возвратов) вследствие нарушения требуемой точности расчетов. Число вычислений правых частей ifu системы (1) можно определить по формуле $ifu = 13 \cdot isa + 12 \cdot iwo$.

Пример 1 [5, с. 145–146].

$$\begin{aligned} y_1' &= 2ty_1y_4, \\ y_2' &= 10ty_1^5y_4, \\ y_3' &= 2ty_4, \\ y_4' &= -2t(y_3 - 1), \\ t &\in [0, 15\pi], \quad y_1(0) = y_2(0) = y_3(0) = y_4(0) = 1, \quad h_0 = 10^{-2}. \end{aligned} \tag{12}$$

Точное решение (12) имеет вид

$$y_1(t) = \exp(\sin t^2), \quad y_2(t) = \exp(5 \sin t^2), \quad y_3(t) = \sin t^2 + 1, \quad y_4(t) = \cos t^2.$$

Целью расчетов было определение влияния неравенства для контроля устойчивости на вычислительные затраты при решении нежестких задач. Из результатов расчетов с точностью $\varepsilon = 10^{-6}$ следует, что при вычислении решения (12) с помощью алгоритма FEL78 без контроля устойчивости затраты $isa = 4055$ и $iwo = 1750$, а для алгоритма FEL78ST — $isa = 4094$, $iwo = 1554$.

Таким образом, для нежестких задач контроль устойчивости практически не влияет на ход вычислений и вычислительные затраты. Отметим, что в построенном алгоритме вопрос о контроле устойчивости можно решить и волевым образом — через формальный параметр при обращении к подпрограмме с реализацией метода.

Пример 2 [10].

$$\begin{aligned} y_1' &= -0.013y_1 - 1000y_1y_3, \\ y_2' &= -2500y_2y_3, \\ y_3' &= -0.013y_1 - 1000y_1y_3 - 2500y_2y_3, \\ t &\in [0, 50], \quad y_1(0) = 1, \quad y_2(0) = 1, \quad y_3(0) = 0, \quad h_0 = 2.9 \cdot 10^{-4}. \end{aligned} \tag{13}$$

Данная задача является жесткой. Из результатов расчетов с точностью $\varepsilon = 10^{-6}$ следует, что при вычислении решения (13) алгоритмом FEL78 без контроля устойчивости затраты $\text{isa} = 37785$, $\text{iwo} = 37752$, $\text{ifu} = 950860$, т.е. почти каждый шаг сопровождается повторным вычислением решения. Это естественно для всех алгоритмов без контроля устойчивости. Для алгоритма FEL78ST затраты таковы: $\text{isa} = 37876$, $\text{iwo} = 454$, $\text{ifu} = 497836$. Из сравнения результатов расчетов следует, что контроль устойчивости приводит к повышению эффективности расчетов почти в два раза. В конце интервала интегрирования фактическая точность вычислений для алгоритма FEL78 на порядок лучше задаваемой, а для алгоритма FEL78ST — на два порядка. При более грубой точности расчетов эффективность FEL78ST возрастает, а для FEL78 остается примерно такой же. Эта тенденция сохраняется при интегрировании всех рассмотренных задач и, в частности, десяти жестких тестовых примеров [10]. Заметим, что для одного из лучших методов — DVERK78 системы Maple 9.5 — на примере (13) затраты $\text{ifu} = 598890$.

Заключение

Использование неравенства для контроля устойчивости фактически не приводит к увеличению вычислительных затрат, потому что оценка максимального собственного числа матрицы Якоби системы (1) осуществляется через ранее вычисленные стадии и не вызывает рост числа вычислений функции f . Как правило, для этих целей применяются три первые стадии. Такая оценка получается грубой. Однако контроль устойчивости в качестве ограничителя на рост шага позволяет избежать негативных последствий грубости оценки. Более того, в некоторых случаях это приводит к нестандартно высокому повышению эффективности алгоритма интегрирования. На участке установления за счет контроля устойчивости старые ошибки стремятся к нулю, а новые невелики из-за малости производных решения. В некоторых случаях вместо оценки максимального собственного числа оценивается следующее по порядку. Шаг интегрирования становится больше максимально допустимого, и с таким шагом интегрирование осуществляется до тех пор, пока не нарушается неравенство для контроля точности. Число таких шагов невелико, однако величина шага может на несколько порядков превышать максимальный шаг по устойчивости. После нарушения неравенства для контроля точности шаг уменьшается до максимально возможного. Такой эффект может повторяться многократно в зависимости от длины участка установления. В результате средний шаг интегрирования может превышать максимально допустимый.

Список литературы

- [1] SHAMPINE L.M. Implementation of Rosenbrock methods // ACM Transaction on Mathematical Software. 1982. Vol. 8, N 5. P. 93–113.
- [2] НОВИКОВ Е.А., НОВИКОВ В.А. Контроль устойчивости явных одношаговых методов интегрирования обыкновенных дифференциальных уравнений // Докл. АН СССР. 1984. Т. 277, № 5. С. 1058–1062.
- [3] НОВИКОВ Е.А. Явные методы для жестких систем. Новосибирск: Наука, 1997. 195 с.
- [4] FENLBERG E. Classical fifth-, sixth-, seventh- and eighth order Runge — Kutta formulas with step size control // Computing. 1969. Vol. 4. P. 93–106.
- [5] ХАЙРЕР Э., НЕРСЕТТ С., ВАННЕР Г. Решение обыкновенных дифференциальных уравнений. Нежесткие задачи: Пер. с англ. М.: Мир, 1990. 512 с.
- [6] МОИСЕЕВ Н.Н. Математические задачи системного анализа. М.: Наука, 1981.
- [7] ЯНЕНКО Н.Н., КАРНАЧУК В.И., КОНОВАЛОВ А.Н. Проблемы математической технологии // Числ. методы механики сплошной среды: Сб. науч. тр. / АН СССР. Сиб. отд-ние ВЦ; ИТПМ. 1977. Т. 8, № 3. С. 129–157.
- [8] ШОРНИКОВ Ю.В. и др. Инструментальные средства машинного анализа: Свидетельство официальной регистрации программы для ЭВМ № 2005610126. М.: Роспатент, 2005.
- [9] БЕНЬКОВИЧ Е.А., КОЛЕСОВ Ю.Б., СЕНИЧЕНКОВ Ю.Б. Практическое моделирование динамических систем. СПб.: БХВ-Петербург, 2002. 464 с.
- [10] ENRIGHT W.H., HULL T.E. Comparing numerical methods for the solutions of systems of ODE's // BIT. 1975. N 15. P. 10–48.

*Поступила в редакцию 22 марта 2006 г.,
в переработанном виде — 3 мая 2006 г.*