УДК 519.64

МАТЕМАТИКА

Член-корреспондент АН Армянской ССР А. Б. Нерсисян

Новые алгоритмы численного решения интегральных уравнений второго рода

(Представлено 10/VII 1989)

В ряде теоретических и прикладных исследований, как классических, так и новейших, возникает задача решения интегрального уравнения Фредгольма второго рода

$$(I - K)y = y(x) - \int_D^x K(x, t)y(t)dt = f(x), \quad (1)$$

где $x \in D \subset R^n$; y, f — вектор-функции размерности $m \geq 1$, K — матрица-функция $(m \times m)$, а оператор K вполне непрерывен в соответствующем пространстве комплекснозначных функций.

Традиционные численные методы решения уравнения (1) основаны на его непосредственном приближенном сведении к линейной алгебраической системе. Основными путями такого сведения являются проекционный метод и метод квадратур (кубатур), причем этот подход хорошо разработан, вплоть до пакетов соответствующих программ (см. (1)).

С другой стороны, в 1954–56 гг. С. Л. Соболев, Н. П. Сергеев и М. Г. Крейн (2–4) заложили основы нового подхода к решению уравнения (1) при $n = 1$, когда D — отрезок (a, b) , изучив оператор $I - K$ на отрезке переменной длины (a, τ) , $(\tau \leq b)$. По сути дела речь идет о вольтерровской факторизации Фредгольмова оператора и о соответствующем „континуальном“ аналоге метода Гаусса. Систематическое исследование этой схемы и ее реализацию в важнейших ситуациях провели А. Мак-Набб и А. Шумицки (5,6). Впоследствии этот же путь избрали Н. Б. Енгибарян и др. (см. (7,8)). Более общий подход, включающий многомерный ($n \geq 2$) случай и невольтерровскую факторизацию, был намечен в (9).

В предлагаемой заметке приводятся результаты альтернативного подхода, позволяющего реализовать здесь основные идеи методов типа Рунге-Кутты и Адамса, относящихся, как известно, к решению обыкновенных дифференциальных уравнений. В случае ядер специальной структуры (см. (10)) на этой основе можно реализовать наиболее экономичные алгоритмы.

1. Основные соотношения. Рассмотрим подробно одномерный вариант уравнения (системы) (1)

$$y(x, \tau) = \int_0^{\tau} K(x, t)y(t, \tau)dt + f(x), \quad (2)$$

где $\tau > 0$ — пока что свободный параметр.

Всюду в дальнейшем будем считать, что ядро K и свободный член f обладают необходимой гладкостью.

Потребовав, чтобы уравнение (2) было однозначно разрешимо при данном τ , легко приходим к следующему соотношению для его резольвентного ядра (см. (2-4)):

$$\frac{\partial}{\partial \tau} R(x, t, \tau) = R(x, \tau, \tau)R(\tau, t, \tau). \quad (3)$$

Заметим теперь, что если ядро K задано в квадрате $(0,1) \times (0,1)$, то для любого $\tau \in (0,1)$, исходя из известных уравнений для резольвенты, можно считать и R определенным в том же квадрате. При таком подходе очевидно, что $R(x, t, 0) = K(x, t)$, $(x, t) \in (0,1) \times (0,1)$, $y(x, 0) = f(x)$, $x \in (0,1)$.

Из этих и аналогичных соображений, рассмотрев значения $\tau = nh, (n+1)h$, приходим к соотношениям (ради удобства в третьем аргументе R множитель h иногда опущен):

$$R(x, t, n+1) = R(x, t, n) + \int_{nh}^{(n+1)h} R(x, s, s)R(s, t, s)ds. \quad (4)$$

$$R(x, t, n+1) = R(x, t, n) + \int_{nh}^{(n+1)h} R(x, s, n)R(s, t, n+1)ds; \quad (5)$$

$$R(x, t, n+1) = R(x, t, n) + \int_{nh}^{(n+1)h} R(x, s, n+1)R(s, t, n)ds. \quad (6)$$

Отметим, что в (4) требуется существование обратного оператора (т. е. ядра R) при всех $\tau \in (nh, (n+1)h)$, в то время как в (5) и (6) — только при двух значениях $\tau = nh, (n+1)h$.

Начнем с задачи рекуррентного вычисления $R(x, t, \tau)$ с указанием сетки по x, t и τ и порядка точности применяемых квадратурных формул.

2. *Одношаговые методы.* Обозначим $h = N^{-1}$ и условимся индексы (x, t) и τ указывать без множителя h . Через E обозначим единицу из C^m .

Алгоритм 1. $x, t = 0, 1, \dots, N; n = 0, 1, \dots, N-1$:

$$R(x, t, n+1) = R(x, t, n) + hR(x, n+1, n)(E - \alpha_n)^{-2}R(n+1, t, n) + O(h^2), \quad (7)$$

где α_n определяется как решение квадратного уравнения

$$h\alpha_n^2 - \alpha_n + R(n+1, n+1, n) = 0. \quad (7')$$

Алгоритм 2. $x, t = 0, 1, \dots, N; n = 0, 1, \dots, N-1$:

$$R(x, t, n+1) = R(x, t, n) + hR(x, n, n)(E - hR(n, n, n))^{-1}R(n, t, n) + O(h^2). \quad (8)$$

Алгоритм 3. $x, t = 0, 1, \dots, N; n = 0, 1, \dots, N-1;$

$$R(x, t, n+1) = R(x, t, n) + \frac{h}{2} \left\{ R(x, n, n)R(n, t, n) + R(x, n+1, n) + \right. \\ \left. + \frac{h}{2} R(x, n, n)R(n, n+1, n) \right\} \left(E - \frac{h}{2} z_n \right)^{-2} \left\{ R(n+1, t, n) + \right. \\ \left. + \frac{h}{2} R(n+1, n, n)R(n, t, n) \right\} + O(h^3), \quad z_n = h(n+1, n+1, n+1). \quad (9)$$

Алгоритм 4. $x, t = 0, 1, \dots, N; n = 0, 1, \dots, N-1;$

$$R(x, t, n+1) = R(x, t, n) + \frac{h}{2} \left\{ R(x, n, n)R(n, t, n+1) + \right. \\ \left. + R(x, n+1, n)R(n+1, t, n+1) \right\} + O(h^3), \quad (10)$$

где величины $R(n, t, n+1)$ и $R(n+1, t, n+1)$ предварительно определяются из системы, получаемой подстановкой в соотношение (10) $x = n, n+1$.

Алгоритм 5. $x, t = 1/2, 3/2, 5/2, \dots, N-1/2; n = 0, 1, \dots, N-1;$

$$R(x, t, n+1) = R(x, t, n) + hR(x, n+1/2, n)(E - hR(n+1/2, n+1/2, n))^{-1}R(n+1/2, t, n) + O(h^3) \quad (11)$$

Алгоритм 6. $x, t = 0, 1/2, 1, 3/2, \dots, N-1/2, N; n = 0, 1, \dots, N-1.$

$$R(x, t, n+1) = R(x, t, n) + \frac{h}{6} \left\{ R(x, n, n)R(n, t, n+1) + 4R(x, n+1/2, n)R(n+1/2, t, n+1) + \right. \\ \left. + R(x, n+1, n)R(n+1, t, n+1) \right\} + O(h^5). \quad (12)$$

3. Многошаговые методы. Остановимся на использовании (4).

Алгоритм 7. Предварительно, скажем, по алгоритму 6, определяется $R(x, t, 1)$, $x, t = 0, 1, 2, \dots, N$. Далее, при $n = 1, 2, \dots, N-1$

$$R(x, t, n+1) = R(x, t, n-1) + \frac{h}{3} \left\{ R(x, n-1, n-1)R(n-1, t, n-1) + \right. \\ \left. + 4R(x, n, n)R(n, t, n) + R(x, n+1, n+1)R(n+1, t, n+1) \right\} + O(h^5). \quad (13)$$

Алгоритм 8. $x, t = 0, 1, \dots, N$. Значения $R(x, t, k)$, $k = 1, 2, 3$ предварительно вычисляются с точностью $O(h^7)$. Далее, при $n = 3, 4, \dots, N-1$

$$R(x, t, n+1) = R(x, t, n-3) + \frac{4h}{90} \left\{ 7R(x, n-3, n-3)R(n-3, t, n-3) + \right. \\ \left. + 32R(x, n-2, n-2)R(n-2, t, n-2) + 12R(x, n-1, n-1)R(n-1, \right. \\ \left. t, n-1) + 32R(x, n, n)R(n, t, n) + 7R(x, n+1, n+1)R(n+1, \right. \\ \left. t, n+1) \right\} + O(h^7). \quad (14)$$

Здесь приведены, в основном, неявные методы, в которых надо предварительно (как в алгоритме 4) найти решение системы небольшого порядка, либо (как в алгоритме 1) найти α_n в качестве корня.

ограниченного при $h \rightarrow 0$. В последнем случае (особенно при $m \geq 2$) решение уравнения для φ_n разумно проводить методом последовательных приближений или разложением по степеням h до требуемого порядка точности. В случае действительных ядер необходимо учитывать только действительные части решения уравнений типа (7').

Сочетание формул типа (4) — (6) приводит к новым алгоритмам. Предлагаемый подход в большей степени учитывает специфику гладких ядер, чем (см. выше) классический, однако в некоторых случаях преимуществом последнего является наличие возможности применения метода выбора ведущего элемента.

Алгоритмы 1—8 позволяют приближенно находить при $\tau=1$ обратный оператор $(I-K)^{-1}$. Другой путь решения уравнения (2) при $\tau=1$ основан на аналогичных формулам (1) — (6) соотношениях, например

$$y(x, n+1) = y(x, n-k) + \int_{(n-k)h}^{(n+1)h} K(x, s, s) y(s, s) ds, \quad k \geq 0. \quad (15)$$

Это обстоятельство позволяет при реализации алгоритмов изучаемого типа в том же вычислительном процессе находить приближенно и $y(x, 1)$.

Подчеркнем также, что для обоснования указанного в алгоритмах 1—8 порядка точности достаточно потребовать, чтобы $K, f \in C^p$, где $p=1; 2; 4; 6$ соответственно в алгоритмах 1,2; 3,4,5; 6,7; 8.

4. *Многомерный случай.* Алгоритмы указанного типа могут быть построены и для решения многомерного уравнения (1). Рассмотрим, например, случай $D = D_\tau = (0, \tau) \times (0, 1)$. Можно показать, что резольвентное ядро $R(x; t; \tau)$ ($x, t \in D_1, 0 \leq \tau \leq 1$) удовлетворяет условию $R(x; t; 0) = K(x, t)$ и интегродифференциальному уравнению

$$\frac{\partial}{\partial \tau} R(x; t; \tau) = \int_0^1 R(x; \tau, s_2; \tau) R(\tau, s_2; t; \tau) ds_2, \quad x, t \in D_1. \quad (16)$$

откуда нетрудно выписать соотношение типа (4). Соотношения типа (5) — (6) также могут быть легко получены. Для получения алгоритмов того или иного порядка остается применить соответствующие кубатурные формулы.

5. *Автоматический выбор шага и порядка.* Важным достоинством предложенного подхода является возможность выбора в процессе счета удобного шага или алгоритма приемлемого порядка. Поясним это.

В алгоритмах 1—6 допускается выбор сколь угодно густой сети по x, t , без изменения шага по τ . Контролируя на каждом n -том этапе величину соответствующей производной функции $R(x, s, n)R(s, t, n+1)$ по s при $nh \leq s \leq (n+1)h$, можно менять величину шага h или порядок алгоритма исходя из желательной величины ошибки. Сравнивая данную ситуацию с классическим случаем задачи Коши для обыкновенного дифференциального уравнения $y' = f(x, y)$, приходим к выводу, что, с одной стороны, здесь эта идея осуществима сложнее вследствие контроля производных не в отдельных точках, а на от-

резках (на достаточно густом множестве точек), но, с другой стороны, ситуация значительно проще из-за квадратичного вида нелинейности.

6. *Экономичные алгоритмы.* Приведенные алгоритмы полностью векторизованы и, при наличии $O(N^2)$ процессоров, допускают эффективное распараллеливание. На однопроцессорных системах они легко могут быть преобразованы в алгоритмы минимальной сложности, если счет в них вести для индексов $x, t \geq n$, а решение $y(x, 1)$ находить не по (15), а посредством прямого и обратного ходов метода Гаусса.

Разумеется, приведенные здесь алгоритмы для уравнения (2) имеют сложность $O(N^3)$ при $N \rightarrow \infty$, однако широкий класс ядер, выделенный в (10), допускает преобразование алгоритмов 1—8 в алгоритмы сложности $O(N^2)$, к тому же эффективно распараллеливаемые при наличии $O(N)$ процессоров.

7. *Численные результаты.* Хорошо известны прикладные характеристики классических подходов к решению уравнения (2) (см. (1)). Из упомянутых во введении работ, рекомендующих метод факторизации, лишь в работе (5) содержатся серьезные численные результаты, подтверждающие эффективность этого подхода.

Алгоритмы 1—8 были всесторонне апробированы численным экспериментом. Приведем некоторые результаты.

Изучалось скалярное уравнение (2) с ядрами K_i и решениями $f_i (i=1, 2)$

$$K_1 = 0, 5e^{-xt}, \quad y_1 \equiv 1; \quad K_2 = 0, 1 \exp 5(x-t)^2, \quad y_2 = \exp(-5x^2). \quad (17)$$

Результаты сведены в таблице.

A	$A_1(h)$		$A_2(h)$		$A_4(h^2)$		$A_6(h^4)$		$A_7(h^4)$	
	δ_1	δ_2								
4	$4,9 \cdot 10^{-2}$	1,3	$3,7 \cdot 10^{-2}$	1,43	$1,1 \cdot 10^{-3}$	$2,5 \cdot 10^{-1}$	$3,6 \cdot 10^{-6}$	$1,7 \cdot 10^{-1}$	$1,3 \cdot 10^{-5}$	$3,2 \cdot 10^{-1}$
8	$2,5 \cdot 10^{-2}$	$5,7 \cdot 10^{-1}$	$1,9 \cdot 10^{-2}$	1,76	$2,8 \cdot 10^{-4}$	$2,6 \cdot 10^{-1}$	$1,5 \cdot 10^{-7}$	$4 \cdot 10^{-2}$	$7,8 \cdot 10^{-7}$	$1,7 \cdot 10^{-1}$
16	$1,2 \cdot 10^{-2}$	$1,8 \cdot 10^{-1}$	$9,6 \cdot 10^{-3}$	$4,5 \cdot 10^{-1}$	$6,8 \cdot 10^{-5}$	$1,3 \cdot 10^{-1}$	$1,5 \cdot 10^{-7}$	$1,4 \cdot 10^{-3}$	$1,5 \cdot 10^{-7}$	$1,9 \cdot 10^{-3}$
32	$6,1 \cdot 10^{-3}$	$1,5 \cdot 10^{-1}$	$4,8 \cdot 10^{-3}$	$2,5 \cdot 10^{-1}$	$1,7 \cdot 10^{-5}$	$1,5 \cdot 10^{-2}$	$1,2 \cdot 10^{-7}$	$8 \cdot 10^{-5}$	$1 \cdot 10^{-7}$	$1,2 \cdot 10^{-3}$
64	$3 \cdot 10^{-3}$	$8 \cdot 10^{-2}$	$2,4 \cdot 10^{-3}$	$1,2 \cdot 10^{-1}$	$4,1 \cdot 10^{-6}$	$3 \cdot 10^{-3}$	$1,3 \cdot 10^{-7}$	$4,5 \cdot 10^{-6}$		$8,7 \cdot 10^{-5}$

Здесь $A_p (p=1, 2, 4, 6, 7)$ — алгоритм p с указанием порядка его точности, δ_i — абсолютная среднеквадратичная ошибка в случае ядра $K_i (i=1, 2)$, $N (= 4, 8, 16, 32, 64)$ — число отрезков равномерного разбиения интервала $(0, 1)$ ($h = \frac{1}{N}$).

Сравнительный анализ с этими и другими результатами показывает, что алгоритмы 1—8 не уступают соответствующим классическим (см. (1)). Алгоритм 4 несколько точнее алгоритма того же порядка точности из (5), где рассматривалось и ядро K_1 . A_7 хорошо согла-