

ВЫЧИСЛИТЕЛЬНАЯ МАТЕМАТИКА

УДК 519.64

А. Б. Гарибян

Быстрые алгоритмы численного решения интегрального уравнения
 второго рода с ядром теплицева типа

(Представлено чл.-корр. НАН Армении А. Б. Нерсисяном 24/III 1993)

Для численного решения интегрального уравнения Фредгольма
 второго рода

$$y(x) = \int_0^1 K(x, t) y(t) dt + f(x), \quad 0 \leq x \leq 1 \quad (1)$$

с ядром теплицева типа

$$\left(\frac{\partial}{\partial x} + \frac{\partial}{\partial t} \right) K(x, t) = \sum_{i=1}^m p_i(x) q_i(t) \quad (2)$$

рассмотрим уравнение (1) на отрезке переменной длины τ ($0 \leq \tau \leq 1$)

$$y(x, \tau) = \int_0^\tau K(x, t) y(t, \tau) dt + f(x), \quad 0 \leq x \leq \tau. \quad (3)$$

Если для данного $\tau \in [0, 1]$ уравнение (3) однозначно разрешимо,
 то существует его резольвентное ядро $R(x, t, \tau)$ ($0 \leq x, t \leq \tau$), удов-
 летворяющее известным соотношениям:

$$R(x, t, \tau) = K(x, t) + \int_0^\tau K(x, s) R(s, t, \tau) ds; \quad (4)$$

$$R(x, t, \tau) = K(x, t) + \int_0^\tau R(x, s, \tau) K(s, t) ds. \quad (5)$$

Заметим, что (см. (1.2)) поскольку ядро K определено во всем квад-
 рате $[0, 1] \times [0, 1]$, формулы (4) и (5) позволяют рассматривать ре-
 зольвентное ядро R при любом τ во всем прямоугольнике $[0, 1] \times$
 $\times [0, 1]$, а решение уравнения (3) $y(x, \tau)$ — при $x \in [0, 1]$. При этом
 R наследует (по x, t) гладкость ядра $K(x, t)$, а y (при гладкости K) —
 гладкость f . При такой трактовке ясно, что

$$R(x, t, 0) = K(x, t), \quad y(x, 0) = f(x), \quad 0 \leq x, t \leq 1.$$

Всюду в дальнейшем будем исходить из равномерного деления отрезка $[0, 1]$ на N частей с шагом $h = 1/N$ ($N > 2$).

Схема решения уравнения (1) для ядер общего вида предполагает последовательное решение уравнений (3) для значений $\tau = 0, h, 2h, \dots, Nh$, что приводит, в конечном счете, к решению уравнения (1) — $y(x) = y(x, Nh)$. При этом используются соотношения резольвентного ядра и решения уравнения (2), определенные во всем квадрате $[0, 1] \times [0, 1]$ (см. (1))^{*}.

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

или

$$R(x, t, n+1) = R(x, t, n) + \int_{nh}^{(n+1)h} R(x, s, n) R(s, t, n+1) ds; \quad (7)$$

$$y(x, n+1) = y(x, n) + \int_{nh}^{(n+1)h} R(x, s, s) y(s, s) ds, \quad (8)$$

или

$$y(x, n+1) = y(x, n) + \int_{nh}^{(n+1)h} R(x, s, n) y(s, n+1) ds. \quad (9)$$

Для ядер теплицева типа из формулы (4) или (5) можно получить рекуррентное соотношение

$$R(x, n+1) = R(x-1, t-1, n) + \int_{nh}^{(n+1)h} \left[\sum_{i=1}^m P_i(s+x-n-1, s) \times \right. \\ \left. \times Q_i(s+t-n-1, s) + R(s+x-n-1, 0, s) R(0, s+t-n-1, s) \right] ds, \quad (10)$$

где

$$P_i(x, \tau) = \int_0^\tau K(x, s) P_i(s, \tau) ds + p_i(x); \quad (11)$$

$$Q_i(x, \tau) = \int_0^\tau Q_i(s, \tau) K(s, x) ds + q_i(x), \quad (12)$$

$$i = 1, \dots, m.$$

* Всюду далее значения $\tau = nh$, а также сетку дискретизации по x, t будем указывать без множителя h .

Для функций $P_i(x, \tau)$ и $Q_i(x, \tau)$, $i = 1, \dots, m$, исходя из формул (11) и (12) можно получить рекуррентные соотношения

$$\begin{cases} P_i(x, n+1) = P_i(x, n) + \int_{nh}^{(n+1)h} R(x, s, s) P_i(s, s) ds \\ Q_i(x, n+1) = Q_i(x, n) + \int_{nh}^{(n+1)h} Q_i(s, s) R(s, x, s) ds \end{cases} \quad (13)$$

и

$$\begin{cases} P_i(x, n+1) = P_i(x, n) + \int_{nh}^{(n+1)h} R(x, s, n) P_i(s, n+1) ds \\ Q_i(x, n+1) = Q_i(x, n) + \int_{nh}^{(n+1)h} Q_i(s, n+1) R(s, x, n) ds. \end{cases} \quad (14)$$

Заметим, что

$$\begin{aligned} R(x, t, 0) &= K(x, t), \quad y(x, 0) = f(x); \\ P_i(x, 0) &= p_i(x), \quad Q_i(x, 0) = q_i(x), \quad i = 1, \dots, m. \end{aligned} \quad (15)$$

В отличие от алгоритмов для ядер общего вида, описанных в (1) и представленных в (3), сложности $O(N^3)$ ($N \rightarrow \infty$), требующих память объема $O(N^2)$, в данном случае на основе формул (8), (10), (13), (15) удастся построить алгоритмы сложности $O(N^2)$ с памятью объема $O(N)$. Общие принципы построения таких алгоритмов изложены в (2).

Если применить к интегральным соотношениям (8), (10) и (13) квадратурную формулу, нетрудно видеть, что достаточно ограничиться рассмотрением следующих значений резольвентного ядра:

$$R(x, 0, \tau), \quad R(0, x, \tau), \quad R(x, \tau, \tau), \quad R(\tau, x, \tau), \quad 0 \leq x, \tau \leq 1. \quad (16)$$

Если обозначить

$$\begin{aligned} \Phi(x, \tau) &= R(0, x, \tau), & \Psi(x, \tau) &= R(x, 0, \tau), \\ \Phi_1(x, \tau) &= R(\tau, x, \tau), & \Psi_1(x, \tau) &= R(x, \tau, \tau), \end{aligned} \quad (17)$$

то в упомянутых алгоритмах будут фигурировать значения $(2m+5)$ векторов.

В качестве примера построим один такой алгоритм, полученный на основе вышеуказанных формул, в которых к интегралам применена квадратурная формула по правилу средней точки

$$\int_a^b \varphi(x) dx = (b-a) \varphi\left(\frac{b+a}{2}\right) + O((b-a)^3). \quad (18)$$

Алгоритм

Рекуррентные соотношения алгоритма

$$y(x, n+1) = y(x, n) + h \Psi_1 \left(x, n + \frac{1}{2} \right) y \left(n + \frac{1}{2}, n + \frac{1}{2} \right),$$

$$x = 0, \frac{1}{2}, 1, \dots, N - \frac{1}{2}, N;$$

$$\Phi_1(x, n+1) = \Phi_1(x-1, n) + h \cdot \left[\sum_{i=1}^m \rho_i \left(n + \frac{1}{2}, n + \frac{1}{2} \right) \times \right. \\ \left. \times Q_i \left(x - \frac{1}{2}, n + \frac{1}{2} \right) + \Psi \left(n + \frac{1}{2}, n + \frac{1}{2} \right) \Phi \left(x - \frac{1}{2}, n + \frac{1}{2} \right) \right];$$

$$\Psi_1(x, n+1) = \Psi_1(x-1, n) + h \sum_{i=1}^m \rho_i \left(x - \frac{1}{2}, n + \frac{1}{2} \right) \times \\ \times Q_i \left(n + \frac{1}{2}, n + \frac{1}{2} \right) + \Psi \left(x - \frac{1}{2}, n + \frac{1}{2} \right) \Phi \left(n + \frac{1}{2}, n + \frac{1}{2} \right);$$

$$x = 1, \frac{3}{2}, 2, \dots, N - \frac{1}{2}, N;$$

$$\rho_i(x, n+1) = \rho_i(x, n) + h \cdot \Psi_1 \left(x, n + \frac{1}{2} \right) \rho_i \left(n + \frac{1}{2}, n + \frac{1}{2} \right);$$

$$Q_i(x, n+1) = Q_i(x, n) + h \cdot \Phi_1 \left(x, n + \frac{1}{2} \right) Q_i \left(n + \frac{1}{2}, n + \frac{1}{2} \right),$$

$$i = 1, \dots, m; \quad x = 0, \frac{1}{2}, 1, \dots, N - \frac{1}{2}, N.$$

Очевидно, что значения $\Phi(x, n+1)$ и $\Psi(x, n+1)$ (т. е. значения $R(0, x, n+1)$ и $R(x, 0, n+1)$) нельзя вычислить, используя формулу (10). В данном алгоритме для вычисления указанных значений употребляется рекуррентная формула для ядер общего вида (6), в которой фигурируют только представленные в (16) значения R . Применяя формулу (18) к данному соотношению, для функций Φ и Ψ получим (с точностью $O(h')$, $h \rightarrow 0$)

$$\Phi(x, n+1) = \Phi(x, n) + h \Phi \left(n + \frac{1}{2}, n + \frac{1}{2} \right) \Phi_1 \left(x, n + \frac{1}{2} \right);$$

$$\Psi(x, n+1) = \Psi(x, n) + h \cdot \Psi_1 \left(x, n + \frac{1}{2} \right) \Psi \left(n + \frac{1}{2}, n + \frac{1}{2} \right),$$

$$x = 0, \frac{1}{2}, 1, \dots, N - \frac{1}{2}, N.$$

Заметим, что $\Phi_1(0, n+1) = \Psi(n+1, n+1)$ и $\Psi_1(0, n+1) = \Phi(n+1, n+1)$, а значения $\Phi_1\left(\frac{1}{2}, n+1\right)$ и $\Psi_1\left(\frac{1}{2}, n+1\right)$ (т. е. значения $R\left(n+1, \frac{1}{2}, n+1\right)$ и $R\left(\frac{1}{2}, n+1, n+1\right)$), которые нельзя вычислить по формуле (6) (число векторов в алгоритме возрастает), будем определять по формуле, аналогичной (10), где интегрирование по τ ведется от $\left(n + \frac{1}{2}\right)h$ до $(n+1)h$. Если к этой формуле применить квадратурную формулу трапеций, получим:

$$\begin{aligned} \Psi_1\left(\frac{1}{2}, n+1\right) &= \Psi_1\left(0, n+\frac{1}{2}\right) + \frac{h}{4} \left[\sum_{i=1}^m \left(P_i\left(0, n+\frac{1}{2}\right) \times \right. \right. \\ &\times Q_i\left(n+\frac{1}{2}, n+\frac{1}{2}\right) + P_i\left(\frac{1}{2}, n+1\right) Q_i(n+1, n+1) + \\ &+ \Psi\left(0, n+\frac{1}{2}\right) \Phi\left(n+\frac{1}{2}, n+\frac{1}{2}\right) + \\ &\left. \left. + \Psi\left(\frac{1}{2}, n+1\right) \Phi(n+1, n+1) \right) \right]; \\ \Phi_1\left(\frac{1}{2}, n+1\right) &= \Phi_1\left(0, n+\frac{1}{2}\right) + \frac{h}{4} \left[\sum_{i=1}^m P_i\left(n+\frac{1}{2}, n+\frac{1}{2}\right) \times \right. \\ &\times Q_i\left(0, n+\frac{1}{2}\right) + P_i(n+1, n+1) Q_i\left(\frac{1}{2}, n+1\right) + \\ &+ \Psi\left(n+\frac{1}{2}, n+\frac{1}{2}\right) \Phi\left(0, n+\frac{1}{2}\right) + \\ &\left. \left. + \Psi(n+1, n+1) \Phi\left(\frac{1}{2}, n+1\right) \right) \right]. \end{aligned}$$

Предлагаемый алгоритм является двухэтапным, так как в рекуррентных формулах фигурируют три значения τ : при вычислении значений функций Φ , Ψ , Φ_1 , Ψ_1 , y , P_i , Q_i , $i = \overline{1, m}$ для $\tau = n+1$ используются значения этих функций для $\tau = n$ и $\tau = n + \frac{1}{2}$. Отсюда следует, что

а) сеть дискретизации по τ должна быть еще гуще: $\tau = 0, \frac{1}{2}, 1, \dots, N$;

б) значения всех функций должны быть заданы для $\tau = 0$ и $\tau = \frac{1}{2}$, а для $\tau = 1, \frac{3}{2}, 2, \dots, N$ — вычислены рекуррентными соотно-

шениями алгоритма. Численным решением уравнения (1) с ядром типа (2) будут значения функции $y(x, N)$, $x = 0, \frac{1}{2}, 1, \dots, N$, полученные в результате описанного процесса.

Формулами (15) определяются начальные значения всех функций для $\tau = 0$:

$$\Phi(x, 0) = \Phi_1(x, 0) = K(0, x); \quad \Psi(x, 0) = \Psi_1(x, 0) = K(x, 0);$$

$$P_i(x, 0) = p_i(x); \quad Q_i(x, 0) = q_i(x), \quad i = 1, \dots, m;$$

$$y(x, 0) = f(x) \quad x = 0, \frac{1}{2}, 1, \dots, N - \frac{1}{2}, N,$$

а для $\tau = \frac{1}{2}$ начальные значения функций Φ , Ψ , Φ_1 , Ψ_1 определяются по формуле (7), P_i , Q_i , $i = 1, \dots, m$ — по формулам (14), y — по формуле (9) с помощью применения к ним квадратурной формулы (18).

$$\Phi\left(x, \frac{1}{2}\right) = \Phi(x, 0) + \alpha \cdot K\left(0, \frac{1}{4}\right) K\left(\frac{1}{4}, x\right);$$

$$\Psi\left(x, \frac{1}{2}\right) = \Psi(x, 0) + \alpha \cdot K\left(\frac{1}{4}, 0\right) K\left(x, \frac{1}{4}\right);$$

$$\Phi_1\left(x, \frac{1}{2}\right) = K\left(\frac{1}{2}, x\right) + \alpha \cdot K\left(\frac{1}{2}, \frac{1}{4}\right) K\left(\frac{1}{4}, x\right);$$

$$\Psi_1\left(x, \frac{1}{2}\right) = K\left(x, \frac{1}{2}\right) + \alpha \cdot K\left(\frac{1}{4}, \frac{1}{2}\right) K\left(x, \frac{1}{4}\right);$$

$$P_i\left(x, \frac{1}{2}\right) = P_i(x, 0) + \alpha \cdot K\left(x, \frac{1}{4}\right) p_i\left(\frac{1}{4}\right);$$

$$Q_i\left(x, \frac{1}{2}\right) = Q_i(x, 0) + \alpha \cdot K\left(\frac{1}{4}, x\right) q_i\left(\frac{1}{4}\right);$$

$$y\left(x, \frac{1}{2}\right) = y(x, 0) + \alpha \cdot K\left(x, \frac{1}{4}\right) f\left(\frac{1}{4}\right);$$

$$\alpha = \frac{h}{4} / \left(1 - \frac{h}{2} K\left(\frac{1}{4}, \frac{1}{4}\right)\right),$$

$$x = 0, \frac{1}{2}, 1, \dots, N - \frac{1}{2}, N, \quad i = 1, \dots, m.$$

Заметим, что сети дискретизации по τ и по x совпадают. Точность алгоритма $16N^2(3 + 2m) + O(N)$, требуемая память — порядка $4N(5 + 2m)$. Алгоритм был реализован в численном эксперименте на

компьютере IBM PC AT, тактовая частота 12 МГц. В частности рассматривались следующие ядра теплицева типа:

$$K_1(x, t) = \begin{cases} \frac{1}{2(x-t)^2} \left(\ln \frac{x+1}{t+1} - \frac{x-t}{x+1} \right), & x \neq t \\ \frac{1}{4(t+1)^2}, & x = t \end{cases};$$

$$K_2(x, t) = \frac{(x+t)(2(x+t)-3)}{6} + xt \left(\frac{2}{3} - t + x(t-1) \right) + \frac{1}{5}.$$

Численные результаты приведены в таблице.

Ядро	δ_{16}	t_{16}	δ_{32}	t_{32}	δ_{64}	t_{64}	δ_{128}	t_{128}
$K_1(x, t)$	7.4E-5	2	1.9E-5	6	5.0E-6	22	1.2E-6	83
$K_2(x, t)$	1.0E-4	2	2.8E-5	8	7.3E-6	29	1.8E-6	115

Здесь δ_N — среднеквадратичная ошибка при дискретизации в N точках, t_N — время работы программы в секундах.

Наблюдения показывают, что построение экономичных алгоритмов для ядер теплицева типа достижимо при использовании разных квадратурных формул. При этом получаются разнообразные алгоритмы разных порядков точности: одношаговые и многошаговые, с равномерными и неравномерными сетями дискретизации по x , с совпадающими и несовпадающими сетями дискретизации по x и по t .

В заключение заметим, что алгоритмы указанного типа абсолютно устойчивы при условии обратимости операторов (3) при всех $0 \leq \tau < 1$ (см. (2)).

Ереванский государственный университет

Հ. Բ. ՂԱՐԻՔՅԱՆ

Տյուպլիցյաճ տիպի կորիզով երկրորդ սեռի ինտեգրալ հավասարման
թվային լուծման առաջ ալգորիթմներ

Երկրորդ սեռի ինտեգրալ հավասարման թվային լուծման դիտարկվող սխեման ենթադրում է (3) տիպի հավասարումների հաջորդաբար լուծում $\tau=0, h, 2h, \dots, Nh$ պրժեքների համար: Ելնելով (3) հավասարման ռեզուլտենտային ֆունկցիայի և լուծման համար հայտնի (6)–(9) առնչություններից, կարելի է մշակել (1) հավասարման թվային լուծման $O(N^3)$ բարդության ալգորիթմներ, որոնց իրականացման համար պահանջվում է $O(N^2)$ մեքենայական հիշողություն: Եթե (1) հավասարման կորիզը տյուպլիցյան տիպի է, ապա (10)–(15) առնչություններից ելնելով, հնարավոր է կառուցել

նույնատիպ ալգորիթմներ, որոնց բարդությունը $O(N^2)$ է, իսկ անհրաժեշտ մեքենայական հիշողությունը՝ $O(N)$: Որպես օրինակ կառուցված է $O(h^2)$ ճշտությամբ երկթայլ ալգորիթմ:

Л И Т Е Р А Т У Р А — Գ Ր Ա Շ Ա Ն Ո Ւ Թ Յ Ո Ւ Ն

1 А. Б. Нерсисян, ДАН АрмССР, т. 89, № 4, с. 171—176 (1989) 2 А. Б. Нерсисян, Изв. АН Армении. Математика, т. 27, № 2 (1992). 3 А. В. Гаспарян, А. Б. Нерсисян, Численное решение интегральных уравнений с автоматическим выбором шага, Ереван. 15 с., Деп. в АрмНИИНТИ 29. 12. 89, № 68—Ар89, 1989.