ALGORITM EFECTIV PENTRU REZOLVAREA ECUAŢIILOR INTEGRALE SINGULARE

Titu CAPCELEA

Catedra Matematica Aplicată

We present a fast algorithm with $O(n\log_2 n)$ computational complexity for the approximate solution of Cauchy singular integral equations with smooth continuous coefficients, defined on the unit circle of the complex plane. In order to obtain this algorithm we adopt Amosov`s idea [1] to the quadrature method. In the case when the approximate method is applicable [2], the solution φ_n of the mentioned fast algorithm satisfies the same convergence rate as the solution φ_n^* of the quadrature method.

Introducere

Fie Γ_0 este circumferinţa unitate a planului complex $\mathbb{C}$ cu centrul în origine, iar $L_2(\Gamma_0)$ este spaţiul tuturor funcţiilor $f:\Gamma_0\to\mathbb{C}$ măsurabile după Lebesgue şi integrabile la pătrat pe Γ_0.

În $L_2(\Gamma_0)$ vom considera ecuaţia integrală singulară

$$(A\varphi\equiv)\, a_0(t)\varphi(t)+\frac{b_0(t)}{\pi i}\int_{\Gamma_0}\frac{\varphi(\tau)}{\tau-t}d\tau+\frac{1}{2\pi i}\int_{\Gamma_0}h(t,\tau)\varphi(\tau)d\tau=f(t),\ \ t\in\Gamma_0, \tag{1}$$

unde $a_0,b_0,f:\Gamma_0\to\mathbb{C}$, $h:\Gamma_0\times\Gamma_0\to\mathbb{C}$ sunt funcţii cunoscute, $a_0,b_0\in C(\Gamma_0)$, $h\in C(\Gamma_0\times\Gamma_0), f\in L_2(\Gamma_0)$, iar $\varphi:\Gamma_0\to\mathbb{C}$ este funcţia necunoscută.

În cele ce urmează este comod de a scrie ecuaţia (1) sub forma echivalentă

$$\left(A\varphi\equiv\right) a(t)(P\varphi)(t)+b(t)(Q\varphi)(t)+(K\varphi)(t)=f(t),\ \ t\in\Gamma_0, \tag{2}$$

unde $a(t)=a_0(t)+b_0(t),\ b(t)=a_0(t)-b_0(t),\ P=(I+S)/2,\ Q=I-P,\ I-$ operatorul unitate, S –operatorul integral singular, iar K este operatorul integral regular.

În [2] a fost demonstrat că inversabilitatea operatorului A, descris de membrul stâng al ecuaţiei (2), implică convergenţa în $L_2(\Gamma_0)$ a metodelor de colocaţii şi de cuadraturi pentru rezolvarea aproximativă a acestei ecuaţii. În vederea obţinerii unor estimaţii efective pentru viteza de convergenţă a soluţiilor aproximative către cea exactă, se poate considera scara spaţiilor Sobolev periodice $H^r(\Gamma_0), r\in\mathbf{R}$, definite ca închiderea mulţimii $C^\infty(\Gamma_0)$ a tuturor funcţiilor infinit diferenţiabile $f:\Gamma_0\to\mathbb{C}$ în raport cu norma $\|f\|_r:=\left(\sum_{k\in\mathbb{Z}}\left(1+|k|\right)^{2r}|c_k(f)|^2\right)^{1/2}$, unde $c_k(f)$ sunt coeficienţii Fourier ai funcţiei f în raport cu sistemul $\{t^k\}$ $(t\in\Gamma_0)$. Fie coeficienţii ecuaţiei (1) $a_0,b_0\in H^q(\Gamma_0),\ q>1/2$, iar partea dreaptă $f\in H^s(\Gamma_0)$, $1/2<s\le q$. Vom considera că nucleul $h(t,\tau)$ este funcţie suficient de netedă, astfel încât operatorul K să fie mărginit ca operator ce acţionează din $L_2(\Gamma_0)$ în $H^q(\Gamma_0)$. La aplicarea metodei de colocaţii (în cazul când se satisfac condiţiile de convergenţă a metodei), pentru soluţia aproximativă $\varphi_n^*(t)=\sum_{k=-n}^{n}\alpha_k t^k$ şi soluţia exactă $\varphi(t)$ a ecuaţiei (1), are loc estimaţia

$$\left\|\varphi-\varphi_n^*\right\|_r\le cn^{r-s}\|f\|_s,\ \forall r\in[0;s], \tag{3}$$

unde c este o constantă ce nu depinde de n şi f. Estimaţia (3) rămâne adevărată şi în cazul metodei de cuadraturi (*a se vedea* [3]). Vom menționa că estimația (3) nu poate fi îmbunătățită după ordinul mărimii n în clasa metodelor de aproximare cu polinoame trigonometrice (*a se vedea* [1]).

Este bine cunoscut că pentru a găsi soluția aproximativă a ecuației (1) de forma polinomului trigonometric φ_n conform căreiva metode direct-aproximative, sunt necesare cel puțin $O(n^3)$ operații aritmetice, ținându-se cont de faptul că problema se reduce la rezolvarea unui sistem de ecuații algebrice lineare de dimensiune $2n+1$, iar acesta, în mod obișnuit, se rezolvă prin metoda Gauss.

În prezenta lucrare se propune un algoritm pentru rezolvarea aproximativă a ecuației (1) esențial mai econom în comparație cu metoda colocațiilor sau cu metoda cuadraturilor mecanice. Acest algoritm permite să se combine estimația asimptotică optimală a vitezei de convergență a aproximațiilor în spațiile Sobolev cu estimația $O(n\log_2 n)$ a numărului de operații aritmetice necesare pentru a calcula cei $2n+1$ coeficienți ai polinomului trigonometric ce aproximează soluția exactă. Volumul de memorie operativă necesar pentru realizarea algoritmului este de $O(n)$, pe când pentru alte metode direct-aproximative această mărime este $O(n^2)$.

Descrierea metodei

Vom ține cont că operatorul integral K este complet continuu în $L_2(\Gamma_0)$. Dacă coeficienții ecuației (2) satisfac condiției $a(t)\neq 0, b(t)\neq 0, t\in\Gamma_0$, atunci operatorul A admite regularizare, iar unul dintre regularizatorii lui A este operatorul $B=a^{-1}P+b^{-1}Q$ (*a se vedea* [4]).

Fiind dat numărul n se va alege $m\in\mathbb{N}$ astfel încât $0<m\leq n$ şi $(2n+1)/(2m+1)\in\mathbb{N}$. Vom nota prin P_n mulțimea polinoamelor trigonometrice de forma $\sum\limits_{k=-n}^{n} r_k t^k$ $(t\in\Gamma_0)$, unde r_k $(k=\overline{-n,n})$ sunt numere complexe arbitrare. La fel, vom considera pe Γ_0 rețelele Δ_n şi Δ_m de puncte echidistante $t_j^{(n)}=\exp\big(2\pi ij/(2n+1)\big), j=\overline{-n,n}$ şi, respectiv, $t_j^{(m)}=\exp\big(2\pi ij/(2m+1)\big), j=\overline{-m,m}$.

Pentru funcția f definită pe Γ_0, integrabilă Riemann, există un singur polinom de interpolare Lagrange $(L_n f)(t)=\sum\limits_{k=-n}^{n}\Lambda_k t^k\in P_n$, $\Lambda_k=(2n+1)^{-1}\sum\limits_{j=-n}^{n} f(t_j)t_j^{-k}$, astfel încât $(L_n f)(t_j)=f(t_j)$ pentru fiecare $j=\overline{-n,n}$. Operatorul L_n este proiectorul de interpolare Lagrange. Introducem şi proiectorul ortogonal P_n, ce asociază funcției $f\in L_2(\Gamma_0)$ polinomul $(P_n f)(t)=\sum\limits_{k=-n}^{n} c_k(f)t^k\in P_n$, unde $c_k(f)$ sunt coeficienții Fourier ai funcției f în raport cu sistemul $\{t^k\}$.

Observăm că proiecția $v=P_m\varphi$ a soluției exacte φ pe subspațiul P_m satisface ecuației

$$Av=f-AQ_m\varphi, \tag{4}$$

în care $Q_m=I-P_m$. Dacă $A_c=aP+bQ$ este partea caracteristică a operatorului A, atunci, deoarece $\|Kt^j\|\to 0$ când $|j|\to\infty$, avem $\|AQ_m\varphi-A_cQ_m\varphi\|=\|KQ_m\varphi\|\leq\sum\limits_{j=-\infty}^{-m-1}|c_j(\varphi)|\|Kt^j\|+$ $+\sum\limits_{j=m+1}^{\infty}|c_j(\varphi)|\|Kt^j\|\to 0, m\to\infty$. Ținând cont de aceasta, vom aproxima ecuația de mai sus cu ecuația $Av^{(1)}=f-A_cQ_m\varphi$. Pentru soluțiile v şi $v^{(1)}$ ale acestora are loc $\|v^{(1)}-v\|\leq\|A^{-1}\|\|AQ_m\varphi-A_cQ_m\varphi\|\to 0$, $m\to\infty$.

Deoarece valorile funcției $A_cQ_m\varphi$ nu sunt cunoscute, vom proceda în modul următor. Ținând cont că operatorul B este regularizator al lui A, uşor se poate arăta că $Q_m Bf = Q_m\varphi + Q_m T\varphi$, unde T – operator complet continuu în $L_2(\Gamma_0)$, iar, în virtutea relației $\|Q_m T\varphi\| \to 0, m \to \infty$, obținem $\|Q_m Bf - Q_m\varphi\| \to 0, m \to \infty$. Funcția Bf o vom aproxima cu polinomul $q_n = L_n BL_n f$. Vom avea $\|q_n - Bf\| \le \|L_n BL_n f - BL_n f\| + \|B\|\|L_n f - f\| \to 0$, $n \to \infty$, şi atunci $\|Q_m q_n - Q_m\varphi\| \le \|Q_m\|\|q_n - Bf\| + \|Q_m Bf - Q_m\varphi\| \to 0, m, n \to \infty$; prin urmare, ultima ecuație poate fi înlocuită cu o ecuație ce o aproximează

$$Av^{(2)} = f - A_c Q_m q_n, \tag{5}$$

iar pentru soluțiile $v^{(1)}$ şi $v^{(2)}$ ale acestora avem $\|v^{(2)} - v^{(1)}\| \le \|A^{-1}\|\|A_c\|\|Q_m q_n - Q_m\varphi\| \to 0$, $m, n \to \infty$.

Pentru soluțiile ecuațiilor (4) şi (5) are loc $\|v^{(2)} - v\| \le \|v^{(2)} - v^{(1)}\| + \|v^{(1)} - v\| \to 0, m, n \to \infty$, de aceea, este rațional să se caute soluția aproximativă a ecuației (5) de forma unui polinom trigonometric din P_m. Ținând cont de aceasta, vom descrie un algoritm efectiv pentru rezolvarea ecuației (2) la baza căruia stă următoarea combinație.

Soluția aproximativă a ecuației (2) se va căuta de forma polinomului

$$\varphi_n(t) = \sum_{k=-n}^{n} c_k(\varphi)t^k \in P_n. \tag{6}$$

Coeficienții Fourier „superiori" c_k $(|k| > m)$ ai polinomului φ_n se calculează nemijlocit cu ajutorul regularizatorului B, utilizând în acest scop formula asimptotică $c_k(\varphi) \sim c_k(Bf)$, $|k| \to \infty$. Ceilalți coeficienți c_k $(|k| \le m)$ se găsesc rezolvând aproximativ ecuația (5) conform metodei cuadraturilor mecanice, cu utilizarea unei rețele de puncte mai rare Δ_m. Deoarece $2n+1$ este divizibil prin $2m+1$, fiecare dintre nodurile sistemului $\left\{t_j^{(m)}\right\}_{j=-m}^{m}$ coincide cu careva nod al sistemului $\left\{t_j^{(n)}\right\}_{j=-n}^{n}$.

Fie $v_m \in P_m$ este soluția aproximativă a ecuației (5), obținută conform metodei cuadraturilor mecanice.

În baza celor exspuse mai sus, soluția aproximativă a ecuației (2) se defineşte de relațiile

$$Q_m\varphi_n = Q_m q_n \tag{7}$$

şi

$$P_m\varphi_n = v_m. \tag{8}$$

Însumând membrii corespunzători ai relațiilor (7) şi (8), obținem

$$\varphi_n = v_m + Q_m q_n. \tag{9}$$

În completă analogie cu cele spuse în [1, p.556-558], se poate arăta că în cazul când $a_0, b_0 \in H^q(\Gamma_0)$, $f \in H^s(\Gamma_0)$, $1/2 < s \le q$, iar $K \in L(L_2(\Gamma_0), H^q(\Gamma_0))$, şi dacă se satisfac condițiile de convergență a metodei de cuadraturi, atunci pentru n suficient de mari şi $m = O(n^{1/3})$ are loc estimația $\|\varphi - \varphi_n\|_r \le c_1 n^{r-s}\|f\|_s$, pentru toți $r > 1/2$, astfel încât $s - (q-s)/2 \le r \le s$ ($\varphi_n(t)$ este aproximația definită prin relația (9)).

Analiza complexității algoritmului

Vom considera că sunt cunoscute valorile $a(t_j^{(n)}), b(t_j^{(n)}), f(t_j^{(n)}),\ j = \overline{-n, n}$ şi $h(t_r^{(m)}, t_s^{(m)}),\ r, s = \overline{-m, m}$. În cele ce urmează vom utiliza matricea Fourier $F_n = \left\{(t_j^{(n)})^k\right\}_{j,k=-n}^{n}$ şi, respectiv, inversa ei $F_n^{-1} = (2n+1)^{-1}\left\{(t_j^{(n)})^{-k}\right\}_{j,k=-n}^{n}$.

Utilizând notația $\omega = e^{(2\pi i)/(2n+1)}$ $(i^2 = -1)$, uşor se poate verifica că coeficienții $\Lambda_k = (2n+1)^{-1}\sum_{j=-n}^{n} f(t_j^{(n)})(t_j^{(n)})^{-k}$ ai polinomului Lagrange $(L_n f)(t) = \sum_{k=-n}^{n}\Lambda_k t^k$ se pot calcula conform relației $\overline{\Lambda} = F_n^{-1}\overline{f}^{(n)}$, unde $\overline{f}^{(n)} = \{f(t_k^{(n)})\}_{k=-n}^{n}$, $\overline{\Lambda} = \{\Lambda_j\}_{j=-n}^{n}$. Calculele din ultima relație se pot realiza în $O(n\log_2 n)$ operații aritmetice, utilizând algoritmul transformării rapide Fourier-TRF (*a se vedea* în [5] algoritmul Cooley-Tukey).

Pentru polinomul $q_n(t) = (L_n B L_n f)(t)$ vom avea $q_n(t) = (L_n\rho)(t) = \sum_{r=-n}^{n}\xi_r t^r$, $\xi_r = (2n+1)^{-1}\sum_{s=-n}^{n}\rho(t_s^{(n)})(t_s^{(n)})^{-r}$, $\rho(t) = (BL_n f)(t) = a^{-1}(t)\sum_{k=0}^{n}\Lambda_k t^k + b^{-1}(t)\sum_{k=-n}^{-1}\Lambda_k t^k$.

Valorile $\rho(t_s^{(n)}), s = \overline{-n,n}$, se vor calcula în $O(n\log_2 n)$ operații aritmetice în modul următor. Se va utiliza algoritmul TRF pentru a calcula valorile $\gamma_s = \sum_{k=0}^{n}\Lambda_k(t_s^{(n)})^k$, $\delta_s = \sum_{k=-n}^{-1}\Lambda_k(t_s^{(n)})^k, s = \overline{-n,n}$, din relațiile $\overline{\gamma} = F_n\overline{\Lambda}^{(1)}$, $\overline{\gamma} = \{\gamma_s\}_{s=-n}^{n}$, $\overline{\Lambda}^{(1)} = (\underbrace{0,\dots,0}_{n},\Lambda_0,\Lambda_1,\dots,\Lambda_n)$ şi, respectiv, $\overline{\delta} = F_n\overline{\Lambda}^{(2)}$, $\overline{\delta} = \{\delta_s\}_{s=-n}^{n}$, $\overline{\Lambda}^{(2)} = (\Lambda_{-n},\dots,\Lambda_{-1}\underbrace{0,\dots,0}_{n+1})$. Apoi, mai sunt necesare $O(n)$ operații pentru a calcula $\rho(t_s^{(n)}) = a^{-1}(t_s^{(n)})\gamma_s + b^{-1}(t_s^{(n)})\delta_s, s = \overline{-n,n}$.

Cunoscând valorile $\rho(t_s^{(n)})$ $(s = \overline{-n,n})$, coeficienții ξ_r $(r = \overline{-n,n})$ se vor calcula din relația $\overline{\xi}^{(n)} = F_n^{-1}\overline{\rho}$, $\overline{\rho} = \{\rho(t_s^{(n)})\}_{s=-n}^{n}$, $\overline{\xi}^{(n)} = \{\xi_j\}_{j=-n}^{n}$, în $O(n\log_2 n)$ operații, utilizând algoritmul TRF.

Astfel, întregul proces de obținere a coeficienților polinomului trigonometric $q_n(t)$ necesită $O(n\log_2 n)$ operații aritmetice.

Pentru a rezolva ecuația (5) conform metodei cuadraturilor mecanice, este necesar să se cunoască valorile părții drepte $g(t) = f(t) - (A_c Q_m q_n)(t)$ în punctele $t_j^{(m)}, j = \overline{-m,m}$. Ținând cont de expresia pentru polinomul $q_n(t)$, vom avea $(A_c Q_m q_n)(t) = a(t)\sum_{r=m+1}^{n}\xi_r t^r + b(t)\sum_{r=-n}^{-m-1}\xi_r t^r$.

Valorile $\zeta_j^{(1)} = \sum_{r=m+1}^{n}\xi_r(t_j^{(m)})^r$, $\eta_j^{(1)} = \sum_{r=-n}^{-m-1}\xi_r(t_j^{(m)})^r, j = \overline{-m,m}$, se vor calcula în $O(n\log_2 n)$ operații în modul următor. Se vor determina vectorii $\zeta = \{\zeta_j\}_{j=-n}^{n}$ şi $\eta = \{\eta_j\}_{j=-n}^{n}$ conform relațiilor $\zeta = F_n\xi^{(1)}$, $\xi^{(1)} = (\underbrace{0,\dots,0}_{n+m+1},\xi_{m+1},\dots,\xi_n)^T$, şi, respectiv, $\eta = F_n\xi^{(2)}$, $\xi^{(2)} = (\xi_{-n},\dots,\xi_{-m-1},\underbrace{0,\dots,0}_{n+m+1})^T$. Deoarece $(2n+1)/(2m+1) \in \mathbb{N}$, atunci pentru $k = \overline{-n,n}, j = \overline{-m,m}$ există exact $2m+1$ valori pentru care $t_k^{(n)} = t_j^{(m)}$, şi atunci, drept urmare, din vectorii ζ şi, respectiv η, se vor selecta câte $2m+1$ elemente, pozițiile de pe care acestea se selectează fiind determinate de acei indici k, pentru care $t_k^{(n)} = t_j^{(m)}$.

Acum, pentru a calcula $g(t_j^{(m)})$ $(j = \overline{-m,m})$, conform relației $g(t_j^{(m)}) = f(t_j^{(m)}) - (a(t_j^{(m)})\zeta_j^{(1)} + b(t_j^{(m)})\eta_j^{(1)})$, mai sunt necesare $O(m)$ operații; astfel, în baza celor expuse, calculul valorilor părții drepte a ecuației (5) necesită în total $O(n\log_2 n)$ operații.

Coeficienţii polinomului $v_m \in P_m$ (soluţia aproximativă a ecuaţiei (5), obţinută conform metodei cuadraturilor mecanice) se pot găsi în $O(m^3)$ operaţii, rezolvând un SEAL de dimensiune $2m+1$ conform metodei Gauss. Ţinând cont de observaţia referitor la calculul valorilor părţii drepte a ecuaţiei (5), numărul operaţiilor aritmetice necesare pentru determinarea soluţiei aproximative v_m este de $O(m^3 + n\log_2 n)$.

Vom alege m astfel, încât $c_1 n^{1/3} \le m \le c_2 n^{1/3}$, unde c_1 şi c_2 sunt constante pozitive ce nu depind de n şi m. Atunci, se poate concluziona că pentru a obţine soluţia aproximativă $\varphi_n(t)$ este necesar un volum de calcule $O(n\log_2 n)$.

Descrierea algoritmului de calcul

Algoritmul metodei descrise mai sus poate fi divizat în cinci paşi.

Pasul 0

Fiind dat numărul $n \in \mathbb{N}$ $(n > 1)$, vom alege $m \in \mathbb{N}$, astfel încât $(2n+1)/(2m+1) \in \mathbb{N}$ şi $m = O(n^{1/3})$.

Pasul 1

Se determină vectorul $\overline{\xi}^{(n)} = \{\xi_k\}_{k=-n}^{n}$ (ale cărui componente sunt coeficienţii polinomului $q_n(t) = \sum\limits_{k=-n}^{n} \xi_k t^k$) conform relaţiei

$$\overline{\xi}^{(n)} = F_n^{-1} A_n^{-1} F_n I_n^{+} F_n^{-1} \overline{f}^{(n)} + F_n^{-1} B_n^{-1} F_n I_n^{-} F_n^{-1} \overline{f}^{(n)}$$

unde $\overline{f}^{(n)} = \{f(t_k^{(n)})\}_{k=-n}^{n}$, $I_n^{+} = \mathrm{diag}(\underbrace{0,\dots 0}_{n},\underbrace{1,\dots 1}_{n+1})$, $I_n^{-} = \mathrm{diag}(\underbrace{1,\dots 1}_{n},\underbrace{0,\dots 0}_{n+1})$, $A_n^{-1} = \mathrm{diag}\,(\overline{a}^{(n)})^{-1}$, $B_n^{-1} = \mathrm{diag}\,(\overline{b}^{(n)})^{-1}$, $(\overline{a}^{(n)})^{-1} = \left\{a^{-1}(t_j^{(n)})\right\}_{j=-n}^{n}$, $(\overline{b}^{(n)})^{-1} = \left\{b^{-1}(t_j^{(n)})\right\}_{j=-n}^{n}$, iar F_n şi F_n^{-1} sunt respectiv, matricea Fourier şi inversa ei.

Pasul 2

Se determină vectorul $\overline{g}^{(m)} = \{g(t_j^{(m)})\}_{j=-m}^{m}$ în modul următor

$$\overline{g}^{(m)} = \overline{f}^{(m)} - (A_m I_{m,n} F_n I_n^{+} + B_m I_{m,n} F_n I_n^{-}) I_n^m \overline{\xi}^{(n)},$$

unde

$A_m = \mathrm{diag}\,(\overline{a}^{(m)}),\ B_m = \mathrm{diag}\,(\overline{b}^{(m)}),\ \overline{a}^{(m)} = \left\{a(t_j^{(m)})\right\}_{j=-m}^{m},\ \overline{b}^{(m)} = \left\{b(t_j^{(m)})\right\}_{j=-m}^{m},\ \overline{f}^{(m)} = \left\{f(t_j^{(m)})\right\}_{j=-m}^{m}$

$I_n^m = \mathrm{diag}(\underbrace{1,\dots,1}_{n-m},\underbrace{0,\dots,0}_{2m+1},\underbrace{1,\dots,1}_{n-m})$, $I_{m,n} = \left\{d_{jk}\right\}$, $d_{jk} = \begin{cases} 1, \text{ dacă } t_j^{(m)} = t_k^{(n)} \\ 0, \text{ în rest} \end{cases}$, $j = \overline{-m,m}$, $k = \overline{-n,n}$.

Pasul 3

Se determină coeficienţii α_k $(k = \overline{-m,m})$ ca soluţie a SEAL

$$a(t_j^{(m)})\sum_{k=0}^{m} \alpha_k (t_j^{(m)})^k + b(t_j^{(m)}) \sum_{k=-m}^{-1} \alpha_k (t_j^{(m)})^k + (2m+1)^{-1} \sum_{k=-m}^{m} \alpha_k \sum_{s=-m}^{m} h(t_j^{(m)}, t_s^{(m)})(t_s^{(m)})^{k+1} = g(t_j^{(m)}),$$

$j = \overline{-m,m}$. SEAL se rezolvă utilizând metoda Gauss.

Pasul 4

Se calculează soluţia aproximativă φ_n a ecuaţiei iniţiale conform relaţiei

$$\varphi_n(t) = \sum_{k=-m}^{m} \alpha_k t^k + \sum_{k=-n}^{-m-1} \xi_k t^k + \sum_{k=m+1}^{n} \xi_k t^k .$$

Referințe:

1. Амосов Б.А. О приближенном решении эллиптических псевдодифференциальных уравнений на гладкой замкнутой кривой // Zeitschrift fur Analysis und ihre Anwendungen. -1990. - Vol.9(6). - P.545-563.
2. Capcelea T. Collocation and quadrature methods for solving singular integral equations with piecewise continuous coefficients // Buletinul Academiei de Ştiinţe a Republicii Moldova. Matematica. - 2006. - Vol.3(52). - P.27-44.
3. Prössdorf S., Silbermann B. Numerical analysis for integral and related operator equations.- Berlin: Akademie Verlag, 1991.- 542 p.
4. Гохберг И.Ц., Крупник Н.Я. Введение в теорию одномерных сингулярных интегральных операторов.- Кишинев: Штиинца, 1973, с.152.
5. Нуссбаумер Г. Быстрое преобразование Фурье и алгоритмы вычисления сверток. - Москва: Радио и связь, 1985.- 248 с.

Prezentat la 23.03.2007