Iterativne metode za linearne sustave - optimalni parametar za SOR

Znanstveno računanje 1 - Završni projektni zadatak

Ante Ćubela

Zagreb, Veljača 2022.

Sadržaj

1 Uvod

U ovom seminaru ćemo za određene linearne sustave aproksimirati optimalnu vrijednost parametra ω za metodu SOR(ω), to jest vrijednost za koju metoda SOR(ω) konvergira "najbrže".
Svi programi su pisani u programskom jeziku C, uz korištenje "F2C", "LAPACK" i "BLAS" paketa. Programe ćemo konstruirati na sljedeći način:

- Za svaki sustav $Ax = b$ s poznatom matricom sustava A ćemo staviti da je rješenje sustava jednadžbi $x_1 = (1, 1, 1, ..., 1)^T$. Vektor b ćemo zatim eksplicitno izračunati koristeći LAPACK funkciju dgemv. Stoga nećemo provjeravati rješenja naših sustava.

- Startna aproksimacija za našu metodu je $x^{(0)} = (0, 0, 0, ..., 0)^T$. Stoga će greška početne aproksimacije biti $||(1, 1, 1, ..., 1)||_\infty = 1$.

- Za sve vrijednosti ω ćemo koristiti jednak broj iteracija i za najbržu konvergenciju ćemo uzeti onu vrijednost ω za koju je norma razlike između aproksimacije y i rješenja x_1 minimalna, to jest, $||e^{(N)}||_\infty = ||y - x_1||_\infty$ je minimalno. U kodu ćemo zapravo uzimati "prosječnu" brzinu konvergencije $p(\omega) = \left(||e^{(N)}||_\infty\right)^{\frac{1}{N}}$ kao brzinu konvergencije.

- Optimalni ω ćemo aproksimirati na 3 decimalna mjesta. Prvo ćemo testirati brzinu konvergencije za vrijednosti $\omega = 1.05 + 0.1i$, za $i \in \{0, 1, 2, ..., 9\}$. Neka je ω_i vrijednost između tih za koju je konvergencija najbrža. Tada ćemo testirati brzinu konvergencije za vrijednosti $\omega = \omega_i - 0.04 + 0.01i$, za $i \in \{0, 1, 2, ..., 9\}$. Neka je ω_{ii} vrijednost između tih za koju je konvergencija najbrža. Za kraj ćemo testirati brzinu konvergencije za vrijednosti $\omega = \omega_{ii} - 0.004 + 0.001i$, za $i \in \{0, 1, 2, ..., 9\}$. Tako smo donekle obuhvatili vrijednosti iz cijelog intervala $(1, 2)$, izvodeći SOR metodu 30 puta. Od zadnjih 10 testiranih vrijednosti ω, onu za koju $||e^{(N)}||_\infty$ minimalno ćemo uzeti kao optimalni ω za dani sustav.

2 Implementacija SOR metode

Ispod je dana funkcija koja izvršava SOR metodu za sustav $Ax = b$. Ulazni parametri metode su: kvadratna matrica A, vektor desne strane b (izračunat eksplicitno tako da rješenje sustava bude $x = (1, 1, 1, ..., 1)^T$), početna iteracija x_0 (u našem slučaju je uvijek $x_0 = \mathbf{0}$), vrijednost parametra ω u SOR metodi, te broj iteracija SOR metode koji želimo izvršiti. U datotekama s ekstenzijom ".h" su implementirani paketi "F2C", "BLAS" i "LAPACK".

```c
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
#include "f2c.h"
#include "fblaswr.h"
```

```c
#include "clapack.h"

doublereal sor_rjesavac (doublereal* A, doublereal* b, doublereal* x_0,
    integer n, doublereal omega, integer br_iter){

integer i, j, iter;
doublereal pom;

doublereal *r=malloc(n*sizeof(doublereal));
doublereal *y=malloc(n*sizeof(doublereal));

for(i=0; i<n; i++){
     y[i]=x_0[i];
}

integer incb=1, incr=1;
dcopy_(&n, b, &incb, r, &incr);

for(iter=0; iter<br_iter; iter++){
     for(i=0; i<n; i++){

          y[i]*=(1-omega);
          pom=r[i];

          for(j=0; j<i; j++){
               pom-=A[i+j*n]*y[j];
          }
          for(j=i+1; j<n; j++){
               pom-=A[i+j*n]*y[j];
          }
          y[i]+=(pom*omega)/A[i+i*n];
     }
}

doublereal n_inf, max=0;
for(i=0; i<n; i++){
     n_inf=abs(y[i]-1);
     if(n_inf>max){
          max=n_inf;
     }
}

doublereal jedan_kroz_N=1.0/br_iter;
doublereal p=pow(max,jedan_kroz_N);
return p;
}
```

Iteracije smo računali po elementima. To jest, $(k+1)$-a iteracija SOR metode je generirana iz k-te iteracije na sljedeći način:

$$x_i^{(k+1)} = (1-\omega)x_i^{(k)} + \frac{\omega}{a_{ii}}\left(b_i - \sum_{j=1}^{i-1} a_{ij}x_j^{(k+1)} - \sum_{j=i+1}^{n} a_{ij}x_j^{(k)}\right), \text{ za } i = 1, 2, ..., n.$$

3 Neki linearni sustavi i traženje optimalnog ω

Prvo ćemo isprobati našu metodu na sljedećem linearnom sustavu iz [1]:

$$\begin{bmatrix} 101 & -4 & 8 & 12 \\ -4 & 20 & -7 & 3 \\ 8 & -7 & 78 & 32 \\ 12 & 3 & 32 & 113 \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} 117 \\ 12 \\ 111 \\ 160 \end{bmatrix}.$$

Egzaktno rješenje tog sustava je $x = (x_1, x_2, x_3, x_4)^T = (1, 1, 1, 1)^T$. Iz prvih deset testiranih vrijednosti ω (kao što je opisano u uvodu) smo dobili da je konvergencija najbrža za vrijednost $\omega = 1.05$. Za drugih deset vrijednosti smo dobili da je konvergencija najbrža za vrijednost $\omega = 1.06$. Za sljedećih deset vrijednosti smo dobili da je konvergencija najbrža za $\omega = 1.056$, pa ćemo tu vrijednost uzeti kao optimalni parametar za dani sustav, to jest $\omega_{opt} \approx 1.056$. U [3] smo za optimalni parametar uzimali vrijednost 1.05, pa smo dobili rezultat kakav smo očekivali. Radili smo 13 iteracija SOR metode, jer je za veći broj iteracija, zbog preciznosti longfloat-a, program vraćao egzaktno rješenje za više različitih parametara ω, pa ne bi mogli naći optimalni među njima. Program koji to testira je priložen kao "*Primjer1.c*".
Od sada pa nadalje ćemo za sve primjere navesti tri dobivene vrijednosti ω koji označavaju iste tri vrijednosti koje smo naveli ovdje i u uvodu. Zadnju od tih vrijednosti uzimamo kao optimalni ω.

Sljedeći primjer kojega ćemo promatrati je linearni sustav gdje matrica sustava ima vrijednosti 2 na glavnoj dijagonali, te -1 na dvije sporedne dijagonale. To jest sljedeću tridijagonalnu matricu:

$$A = \begin{bmatrix} 2 & -1 & 0 & ... & & & \\ -1 & 2 & -1 & 0 & & & \\ 0 & -1 & 2 & -1 & & & \\ & & ... & ... & ... & & \\ & & & -1 & 2 & -1 \\ & & & & -1 & 2 \end{bmatrix}, \qquad \textbf{(I)}$$

gdje je A kvadratna matrica. Tražili smo optimalni parametar ω za takve matrice dimenzija 100×100, 50×50, te 25×25. Neka je d dimenzija matrice A. Dobili smo sljedeće rezultate za optimalni parametar ω:

$$d = 100 \implies 1.95,\ 1.94,\ 1.939 \implies \boxed{\omega_{opt} \approx 1.939};$$
$$d = 50 \implies 1.85,\ 1.89,\ 1.889 \implies \boxed{\omega_{opt} \approx 1.889};$$
$$d = 25 \implies 1.85,\ 1.81,\ 1.806 \implies \boxed{\omega_{opt} \approx 1.806}.$$

Broj iteracija je 100 za sva tri primjera.

Također smo isprobali primjer iz [3] gdje je dimenzija matrice 99×99, te umjesto 2, na dijagonali imamo vrijednosti 1.9999. Radili smo 100 iteracija metode. Dobijemo sljedeća tri rezultata: 1.95, 1.96, 1.956. Taj rezultat odgovara rezultatu za optimalni ω dan u [3] (1.95). Kod za taj primjer je priložen kao "*Primjer3.c*".

Matrici A iz (**I**) ćemo dodati vrijednosti $\frac{1}{2}$ na sljedeće dvije po redu dijagonale, te ponoviti prethodni postupak za linearni sustav gdje je matrica sustava takva petero-dijagonalna. Dobijemo sljedeće rezultate:

$$\begin{aligned} d = 100 &\implies 1.15,\ 1.12,\ 1.118 \implies \boxed{\omega_{opt} \approx 1.118}; \\ d = 50 &\implies 1.15,\ 1.11,\ 1.112 \implies \boxed{\omega_{opt} \approx 1.112}; \\ d = 25 &\implies 1.15,\ 1.13,\ 1.129 \implies \boxed{\omega_{opt} \approx 1.129}. \end{aligned}$$

Zbog istih razloga kao prije smo koristili različite brojeve iteracija (točnost longfloat-a u jeziku C). Kod je priložen kao "*Primjer4.c*".

4 Diskretizacija Poissonove jednadžbe

Diskretizacijom Poissonove jednadžbe dobijemo sljedeću blok-dijagonalnu matricu dimenzija $(n^2) \times (n^2)$:

$$A = \begin{bmatrix} D & -I & 0 & ... & & \\ -I & D & -I & 0 & & \\ 0 & -I & D & -I & & \\ & & ... & ... & ... & \\ & & & -I & D & -I \\ & & & & -I & D \end{bmatrix}, \qquad \textbf{(II)}$$

gdje je D matrica dimenzija $n \times n$ s vrijednostima 4 na glavnoj dijagonali i -1 na dvije sljedeće dijagonale, a $I = I_n$ jedinična matrica dimenzija $n \times n$.

Kada primjenjujemo SOR metodu na matrice tipa (**II**), poznato je dosta teorijskih rezultata. Za matrice tog tipa je poznato da je optimalni parametar ω za SOR metodu:

$$\omega_{opt} \approx \frac{2}{1 + \sin\frac{\pi}{N+1}},$$

za dovoljno velike vrijednosti N, gdje je N vodeća dimenzija blokova D. Također znamo i da je najmanji spektralni radijus jednak:

$$\rho\left(\mathrm{SOR}_{\omega_{opt}}\right) \approx \frac{1 - \sin\frac{\pi}{N+1}}{1 + \sin\frac{\pi}{N+1}},$$

za dovoljno velike vrijednosti N. Još jedan značajan rezultat jest taj se optimalni parametar ω za SOR metodu ne mijenja ukoliko promatramo matricu (**II**) s "crveno-crnom" numeracijom čvorova. Više o tim teorijskim rezultatima može se naći u [2] i [1].

Sljedeći dio koda iz matrice A dimenzija $n \times n$ generira njezinu "crveno-crnu" permutaciju CC:

```
doublereal* CC=malloc(n*n*sizeof(doublereal));
integer pola=floor((n+1)/2);

for(i=0; i<n; i++){
    for(j=0; j<n; j++){
        if(i%2==0){
            if(j%2==0){
                CC[i/2+(j/2)*n]=A[i+j*n];
            }
            if(j%2){
                CC[i/2+(pola+(j-1)/2)*n]=A[i+j*n];
            }
        }
        if(i%2){
            if(j%2){
                CC[pola+(i-1)/2+(pola+(j-1)/2)*n]=A[i+j*n];
            }
            if(j%2==0){
                CC[pola+(i-1)/2+(j/2)*n]=A[i+j*n];
            }
        }
    }
}
```

Prvo smo tražili optimalni parametar ω za matricu (**II**) dimenzija 81×81 (blokovi D su dimenzija 9×9). Za svaki testirani parametar smo napravili 55 iteracija SOR metode. Kao rezultate smo dobili vrijednosti 1.55, 1.53, 1.529, pa za optimalni parametar ω uzimamo $\omega_{opt} \approx 1.529$. Ovo odgovara našem rezultati iz [3], gdje je dano da je optimalni parametar ω približno 1.53. Također možemo vidjeti da je:

$$\frac{2}{1 + \sin\frac{\pi}{9+1}} \approx 1.528,$$

pa se zbog dovoljno velike dimenzije N naši rezultati poklapaju s teorijskim rezultatima. Program koji ovo testira je priložen kao "*Primjer5.c*".

Zatim smo aproksimirali optimalni parametar ω za matrice **(II)** različitih dimenzija, te smo za svaku od tih matrica napravili "crveno-crnu" permutaciju, i tražili optimalni parametar ω za te matrice. Za matrice dimenzija 81×81 ($N = \sqrt{81} = 9$), 256×256, te

425×425, smo dobili sljedeće rezultate (uz prije spomenute oznake):

$$\begin{aligned}
N = 9 &\implies 1.55, 1.53, 1.530 \quad \text{(za matricu } A\text{)}\\
&\implies 1.55, 1.53, 1.534 \quad \text{(za matricu } CC\text{);}\\
N = 16 &\implies 1.75, 1.71, 1.706 \quad \text{(za matricu } A\text{)}\\
&\implies 1.75, 1.71, 1.706 \quad \text{(za matricu } CC\text{);}\\
N = 25 &\implies 1.75, 1.75, 1.747 \quad \text{(za matricu } A\text{)}\\
&\implies 1.75, 1.75, 1.747 \quad \text{(za matricu } CC\text{).}
\end{aligned}$$

Sve završne aproksimacije su jako slične, pa ćemo uzeti prosjeke optimalnih vrijednosti parametra za matrice A i CC, te provjeriti kako odgovaraju teorijskim rezultatima:

$$\begin{aligned}
N = 9 &\implies \omega_{opt} \approx 1.532 \approx 1.528 \approx \frac{2}{1 + \sin\frac{\pi}{N+1}};\\
N = 16 &\implies \omega_{opt} \approx 1.706 \approx 1.690 \approx \frac{2}{1 + \sin\frac{\pi}{N+1}};\\
N = 25 &\implies \omega_{opt} \approx 1.747 \approx 1.785 \approx \frac{2}{1 + \sin\frac{\pi}{N+1}}.
\end{aligned}$$

Rezultati ne odgovaraju savršeno teorijskim očekivanjima, ali zbog ograničene preciznosti programskog jezika C, smo ujedno bili i ograničeni s brojem iteracija metode koji možemo izvršiti. Dimenzija N također nije dovoljno velika kako bi optimalni parametar savršeno zadovoljio teorijska očekivanja.

Za $N = 9$ smo u 50 iteracija SOR metode s optimalnim parametrom ω došli do rješenja s točnošću:

$$\begin{aligned}
&2.66 \cdot 10^{-13} \text{ za matricu } A;\\
&9.13 \cdot 10^{-14} \text{ za matricu } CC.
\end{aligned}$$

Za $N = 16$ smo u 100 iteracija SOR metode s optimalnim parametrom ω došli do rješenja s točnošću:

$$\begin{aligned}
&6.46 \cdot 10^{-14} \text{ za matricu } A;\\
&3.66 \cdot 10^{-15} \text{ za matricu } CC.
\end{aligned}$$

Za $N = 25$ smo u 120 iteracija SOR metode s optimalnim parametrom ω došli do rješenja s točnošću:

$$\begin{aligned}
&3.06 \cdot 10^{-14} \text{ za matricu } A;\\
&8.44 \cdot 10^{-15} \text{ za matricu } CC.
\end{aligned}$$

Uz očekivani rezultat da će matrice manjih dimenzija konvergirati brže (manje iteracija je potrebno za istu točnost rješenja), također možemo vidjeti da je konvergencija malo bolja za matrice s "crveno-crnim" poretkom, nego za matrice oblika (**II**). Programi za ta tri spomenuta primjera su priloženi kao "*Primjer6.c*", "*Primjer7.c*", te "*Primjer8.c*" respektivno.

5 Izvori pri pisanju rada

[1] Nela Bosner. *25. predavanje "Numerička analiza" na doktorskom studiju 2011./12.*

[2] Nela Bosner. *Dodaci - blok iterativne metode za linearne sustave.*

[3] Nela Bosner. *Iterativne metode za linearne sustave.*