Rozwiązywanie układu równań liniowych metodą Gaussa-Seidla

1. Zastosowanie

Procedura $iZGaussSeidel$ rozwiązuje układ równań liniowych postaci

$$\boldsymbol{Ax} = \boldsymbol{b}\ (1),$$

gdzie $\boldsymbol{A}$ oznacza macierz kwadratową stopnia n, a $x, b\ \in R^n$, metodą Gaussa-Seidla.

2. Opis metody

Macierz $\boldsymbol{A}$ układu (1) jest przekształcana na sumę trzech macierzy, tj.

$$\boldsymbol{A} = \boldsymbol{L} + \boldsymbol{D} + \boldsymbol{U},$$

gdzie $\boldsymbol{L}$ oznacza macierz trójkątną dolną, $\boldsymbol{D}$ − macierz diagonalną, a $\boldsymbol{U}$ oznacza macierz trójkątną górną. Uwzględniając rozkład macierzy $\boldsymbol{A}$, układ równań (1) można zapisać w postaci

$$(\boldsymbol{L} + \boldsymbol{D} + \boldsymbol{U})\boldsymbol{x} = \boldsymbol{b},$$

skąd

$$(\boldsymbol{L} + \boldsymbol{d})\boldsymbol{x} = \ -\boldsymbol{U}\boldsymbol{x} + \boldsymbol{b}.$$

Z powyższej zależności wynika następujący proces iteracyjny:

$$(\boldsymbol{L} + \boldsymbol{D})\boldsymbol{x^{k+1}} = \ -\boldsymbol{U}\boldsymbol{x^k} + \boldsymbol{b},$$

tj.

$$\boldsymbol{x^{k+1}} = -(\boldsymbol{L} + \boldsymbol{D})^{-1}\boldsymbol{U}\boldsymbol{x^k} + (\boldsymbol{L} + \boldsymbol{D})^{-1}\boldsymbol{b}.\ (2)$$

Jeżeli promień spektralny macierzy $-(\boldsymbol{L} + \boldsymbol{U})^{-1}\boldsymbol{U}$ jest mniejszy od 1, to proces iteracyjny (2) jest zbieżny. Z zależności (2) wynika, że $(k+1)$-sze przybliżenie $i - tej$ składowej rozwiązania jest określone wzorem

$$\boldsymbol{x_i^{k+1}} = \frac{-\sum_{j=1}^{i-1}\boldsymbol{a_{ij}}\boldsymbol{x_j^{k+1}} - \sum_{j=i+1}^{n}\boldsymbol{a_{ij}}\boldsymbol{x_j^k} + \boldsymbol{b_i}}{\boldsymbol{a_{ii}}}, i = 1, 2, \dots, n, (3)$$

przy czym $a_{ii} \neq 0$. Proces iteracyjny kończy się, gdy

$$\frac{\left\|\boldsymbol{x^{k+1}} - \boldsymbol{x^k}\right\|}{\boldsymbol{\max}(\|\boldsymbol{x^{k+1}}\|, \|\boldsymbol{x^k}\|)} \leq \ \boldsymbol{\varepsilon}, x^{k+1}\ \neq 0\ lub\ x^k \neq 0,$$

gdzie

$$\|\boldsymbol{x}\| = \boldsymbol{max_{1 \leq i \leq n}}|\boldsymbol{x_i}|,$$

a ε oznacza zadaną dokładność, lub gdy $x^{k+1} = x^k\ = 0$ lub też, gdy liczba iteracji w procesie (3) jest większa od przyjętej wartości maksymalnej.

3. Wywołanie procedury

$$iZGaussSeidel(n, a, b, mit, eps, x, it, st)$$

4. Dane

n – liczba równań (równa liczbie niewiadomych),

a – tablica zawierająca wartości elementów macierzy $\boldsymbol{A}$ (element $a[i,j]$ powinien zawierać wartość a_{ij}, gdzie $i,j = 1,2,\dots,n$),

b – tablica zawierająca wartości składowych wektora $\boldsymbol{b}$ (element $b[i]$ powinien zawierać wartość b_i, gdzie $i = 1,2,\dots,n$);

mit – maksymalna liczba iteracji w procesie (3),

eps – względna dokładność rozwiązania,

x – tablica zawierająca początkowe przybliżenia wartości x_i $(i = 1,2,\dots,n)$.

Uwaga:
Po wykonaniu procedury $iZGaussSeidel$ wartości elementów tablicy x są zmienione.

5. Wyniki

xi – tablica zawierająca rozwiązanie (element $x[i]$ zawiera wartość $x_i, i = 1,2,\dots,n$),

it – liczba iteracji wykonanych w procesie (3).

6. Inne parametry

st – zmienna, której w procedurze $iZGaussSeidel$ przypisuje się jedną z następujących wartości:

- 1, jeżeli $n < 1$,
- 2, gdy macierz A jest osobliwa,
- 3, jeżeli wymagana dokładność rozwiązania nie jest osiągnięta po mit iteracjach,
- 0, w przeciwnym wypadku.

Uwaga:
Jeżeli $st = 1\ lub\ 2$, to po wykonaniu procedury $iZGaussSeidel$ elementy tablicy x nie są zmienione. Gdy $st = 3$, to x zawiera ostatnio obliczone przybliżenie rozwiązania.

7. Typy parametrów

$\boldsymbol{Integer}: it, mit, n, st$
$\boldsymbol{Extended}: interval$
$\boldsymbol{imatrix}: a$
$\boldsymbol{ivector}: b, x$

8. Identyfikatory nielokalne

$\boldsymbol{interval}$ – nazwa typu rekordowego postaci:

$\boldsymbol{type}\ interval = record$
$\boldsymbol{var}\ a, b : Extended;$

Rekord zawiera przeciążone operatory, procedury oraz funkcje dotyczące obliczeń na arytmetyce przedziałowej. Szczegóły implementacji zawarte w pliku $IntervalArithmetic32and64.pas$

ivector – nazwa typu tablicowego $[q_1 .. q_n]$ o elementach typu ***interval***, gdzie $q_1 \leq 1$ oraz $q_n \geq n$,

imatrix – nazwa typu tablicowego $[q_1 .. q_n, \ q_1 .. q_n]$ o elementach typu ***interval***, przy czym $q_1 \leq 1$ i $q_n \geq n$,

9. Kod źródłowy

```
1.  procedure iZGaussSeidel(n: Integer; var a: imatrix; var b: ivector;
2.    mit: Integer; eps: interval; var x: ivector; var it, st: Integer);
3.  var
4.    i, ih, k, kh, khh, lz1, lz2: Integer;
5.    max, r: interval;
6.    cond: Boolean;
7.    x1: ivector;
8.    _0, _1: interval;
9.  begin
10.
11.   _0 := int_read('0');
12.   _1 := int_read('1');
13.   SetLength(x1, n);
14.   if n < 1 then
15.     st := 1
16.   else
17.   begin
18.     st := 0;
19.     cond := true;
20.     for k := 1 to n do
21.       x1[k - 1] := _0;
22.     repeat
23.       lz1 := 0;
24.       khh := 0;
25.       for k := 1 to n do
26.       begin
27.         lz2 := 0;
28.         if a[k - 1, k - 1] = _0 then
29.         begin
30.           kh := k;
31.           for i := 1 to n do
32.             if a[i - 1, k - 1] = _0 then
33.               lz2 := lz2 + 1;
34.           if lz2 > lz1 then
35.           begin
36.             lz1 := lz2;
37.             khh := kh
38.           end
39.         end
40.       end;
41.       if khh = 0 then
42.         cond := false
43.       else
44.       begin
45.         max := _0;
46.         for i := 1 to n do
47.         begin
48.           r := iabs(a[i - 1, khh - 1]);
49.           if (r > max) and (x1[i - 1] = _0) then
50.           begin
51.             max := r;
52.             ih := i
53.           end
54.         end;
55.         if max = _0 then
56.           st := 2
57.         else
```

```
58.         begin
59.           for k := 1 to n do
60.           begin
61.             r := a[khh - 1, k - 1];
62.             a[khh - 1, k - 1] := a[ih - 1, k - 1];
63.             a[ih - 1, k - 1] := r
64.           end;
65.           r := b[khh - 1];
66.           b[khh - 1] := b[ih - 1];
67.           b[ih - 1] := r;
68.           x1[khh - 1] := _1;
69.         end
70.       end;
71.     until not cond or (st = 2);
72.     if not cond then
73.     begin
74.       it := 0;
75.       repeat
76.         it := it + 1;
77.         if it > mit then
78.         begin
79.           st := 3;
80.           it := it - 1
81.         end else begin
82.           for i := 1 to n do
83.           begin
84.             r := b[i - 1];
85.             for k := 1 to i - 1 do
86.               r := r - a[i - 1, k - 1] * x[k - 1];
87.             for k := i + 1 to n do
88.               r := r - a[i - 1, k - 1] * x1[k - 1];
89.             x1[i - 1] := r / a[i - 1, i - 1]
90.           end;
91.           cond := true;
92.           i := 0;
93.           repeat
94.             i := i + 1;
95.             max := iabs(x[i - 1]);
96.             r := iabs(x1[i - 1]);
97.             if max < r then
98.               max := r;
99.             if NotZero(max) then
100.              if iabs(x[i - 1] - x1[i - 1]) / max >= eps then
101.                cond := false;
102.          until (i = n) or not cond;
103.          for i := 1 to n do
104.            x[i - 1] := x1[i - 1]
105.        end;
106.      until (st = 3) or cond;
107.    end
108.  end
109.end;
```

110. Przykłady

Przykład I

Dane:

n=2;

$a[1,1].a := 3,\quad a[1,1].b := 3,\quad a[1,2].a := 2,\quad a[1,2].b := 2,$
$a[2,1].a := 2,\quad a[2,1].b := 2,\quad a[2,2].a := 6,\quad a[2,2].b := 6,$
$b[1].a := 1,\quad b[1].b := 1,\quad b[2].b := 1,\quad b[2].a := 1,$
$x[1].a := 8,\quad x[1].b := 8,\quad x[2].a := 10,\quad x[2].b := 10,$

$mit := 20, \quad eps = 1e-3,$

***Wyniki*:**

$x[1].a := 2.0003612818732465E+0000, \quad x[1].b := 2.0003612818732466E+0000,$
$x[2].a := 1.0001806409366232E+0000, \quad x[2].b := 1.0001806409366233E+0000,$
$st = 0, \quad it = 11$

Przykład II

***Dane*:**

n=4;

$a[1,1].a := 0, \quad a[1,1].b := 0, \quad a[1,2].a := 0, \quad a[1,2].b := 0,$
$a[1,3].a := 1, \quad a[1,3].b := 1, \quad a[1,4].a := 2, \quad a[1,4].b := 2,$
$a[2,1].a := 2, \quad a[2,1].b := 2, \quad a[2,2].a := 1, \quad a[2,2].b := 1,$
$a[2,3].a := 0, \quad a[2,3].b := 0, \quad a[2,4].a := 2, \quad a[2,4].b := 2,$
$a[3,1].a := 7, \quad a[3,1].b := 7, \quad a[3,2].a := 3, \quad a[3,2].b := 3,$
$a[3,3].a := 0, \quad a[3,3].b := 0, \quad a[3,4].a := 1, \quad a[3,4].b := 1,$
$a[4,1].a := 0, \quad a[4,1].b := 0, \quad a[4,2].a := 5, \quad a[4,2].b := 5,$
$a[4,3].a := 0, \quad a[4,3].b := 0, \quad a[4,4].a := 0, \quad a[4,4].b := 0,$
$b[1].a := 1, \quad b[1].b := 1, \quad b[2].b := 1, b[2].a := 1,$
$b[3].a := 1, \quad b[3].b := 1, \quad b[4].a := 1, \quad b[4].b := 1,$
$x[1].a := 0, x[1].b := 0, x[2].a := 0, x[2].b := 0,$
$x[3].a := 0, x[3].b := 0, x[4].a := 0, \quad x[4].b := 0,$
$mit := 100, \quad eps = 1e-14,$

***Wyniki*:**

$x[1].a := -1.1616451848058977E-0020, \quad x[1].b := 7.7443012320393174E-0021,$
$x[2].a := 1.9999999999999999E-0001, \quad x[2].b := 2.0000000000000001E-0001,$
$x[3].a := 1.9999999999999999E-0001, \quad x[3].b := 2.0000000000000001E-0001,$
$x[4].a := 3.9999999999999999E-0001, \quad x[4].b := 4.0000000000000001E-0001,$
$st = 0, \quad it = 46$

Przykład III

***Dane*:**

n=4;

$a[1,1].a := 0, \quad a[1,1].b := 0, \quad a[1,2].a := 0, \quad a[1,2].b := 0,$
$a[1,3].a := 1, \quad a[1,3].b := 1, \quad a[1,4].a := 2, \quad a[1,4].b := 2,$
$a[2,1].a := 2, \quad a[2,1].b := 2, \quad a[2,2].a := 1, \quad a[2,2].b := 1,$
$a[2,3].a := 0, \quad a[2,3].b := 0, \quad a[2,4].a := 2, \quad a[2,4].b := 2,$
$a[3,1].a := 7, \quad a[3,1].b := 7, \quad a[3,2].a := 3, \quad a[3,2].b := 3,$
$a[3,3].a := 0, \quad a[3,3].b := 0, \quad a[3,4].a := 1, \quad a[3,4].b := 1,$
$a[4,1].a := 0, \quad a[4,1].b := 0, \quad a[4,2].a := 5, \quad a[4,2].b := 5,$
$a[4,3].a := 0, \quad a[4,3].b := 0, \quad a[4,4].a := 0, \quad a[4,4].b := 0,$
$b[1].a := 1, \quad b[1].b := 1, \quad b[2].b := 1, b[2].a := 1,$
$b[3].a := 1, \quad b[3].b := 1, \quad b[4].a := 1, \quad b[4].b := 1,$
$x[1].a := 0, x[1].b := 0, x[2].a := 0, x[2].b := 0,$
$x[3].a := 0, x[3].b := 0, x[4].a := 0, \quad x[4].b := 0,$
$mit := 30, \quad eps = 1e-14,$

***Wyniki*:**

$x[1].a := -2.1063457742631235E-0014, \quad x[1].b := -2.1063430637576921E-0014,$

$x[2].a := 1.9999999999999999E-0001,$ $x[2].b := 2.0000000000000001E-0001,$
$x[3].a := 1.9999999999970511E-0001,$ $x[3].b := 1.9999999999970512E-0001,$
$x[4].a := 3.9999999999999999E-0001,$ $x[4].b := 4.0000000000000001E-0001,$
$st = 3,$ $it = 30$