4 Gauss-Jordanova metoda eliminacija

4.1 Podsjetnik

4.1.1 Matrice i osnovne operacije

Matricu A tipa $m \times n$ zapisujemo kao uređenu tablicu elemenata posloženih u m redaka i n stupaca,

$$\mathrm{A} = \begin{bmatrix} a_{11} & a_{12} & a_{13} & \dots & a_{1n} \\ a_{21} & a_{22} & a_{23} & \dots & a_{2n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & a_{m3} & \dots & a_{mn} \end{bmatrix}, \tag{1}$$

a zbog praktičnosti, često se koristi i simbolički zapis

$$\mathrm{A} = [a_{ij}] \tag{2}$$

ili $\mathrm{A} = (a_{ij})$. U programskom jeziku C najjednostavnije je matricu deklarirati kao dvodimenzionalno polje (niz)

$$tip \;\; a[m+1][n+1]; \tag{3}$$

da ne moramo usklađivati indekse obzirom na uobičajeni matematički zapis te obavit inicijalizaciju

$$\begin{aligned} &\forall i \in [1, m] \\ &\quad \forall j \in [1, n] \\ &\qquad a[i][j] = a_{ij}; \end{aligned} \tag{4}$$

odnosno ignorirati elemente polja gdje je $i = 0$ ili $j = 0$. Radi jednostavnosti u tekstu prilikom pisanja algoritama koristimo uobičajenu matematičku notaciju u kojoj a_{ij} predstavlja element matrice $[a_{ij}]$ u i-tom retku i j-tom stupcu.

Ako je A nulmatrica, tada su svi

$$a_{ij} = 0\,, \tag{5}$$

a ako je A kvadratna ($m = n$) matrica reda n, $\mathrm{A} \in \mathscr{M}_n$, koja je:

- jedinična, svi
$$a_{ij} = \delta_{ij}\,, \tag{6}$$
- dijagonalna,
$$a_{ij} = 0 \text{ za } j \neq i\,, \tag{7}$$
- gornja trokutasta,
$$a_{ij} = 0 \text{ za } i < j\,, \tag{8}$$
- donja trokutasta,
$$a_{ij} = 0 \text{ za } i > j\,, \tag{9}$$

Algoritam osnovnih operacija s matricama: $C = A \pm B$, $D = \beta \cdot B$, $T = A^T$

zadani β, a_{ij}, b_{ij}

$\forall i \in [1, m]$

  $\forall j \in [1, n]$

    $c_{ij} = a_{ij} \pm b_{ij}$

    $d_{ij} = \beta \cdot b_{ij}$

    $t_{ij} = a_{ji}$

Algoritam množenja matrica $C = A \cdot B$

zadani a_{ij}, b_{ij}

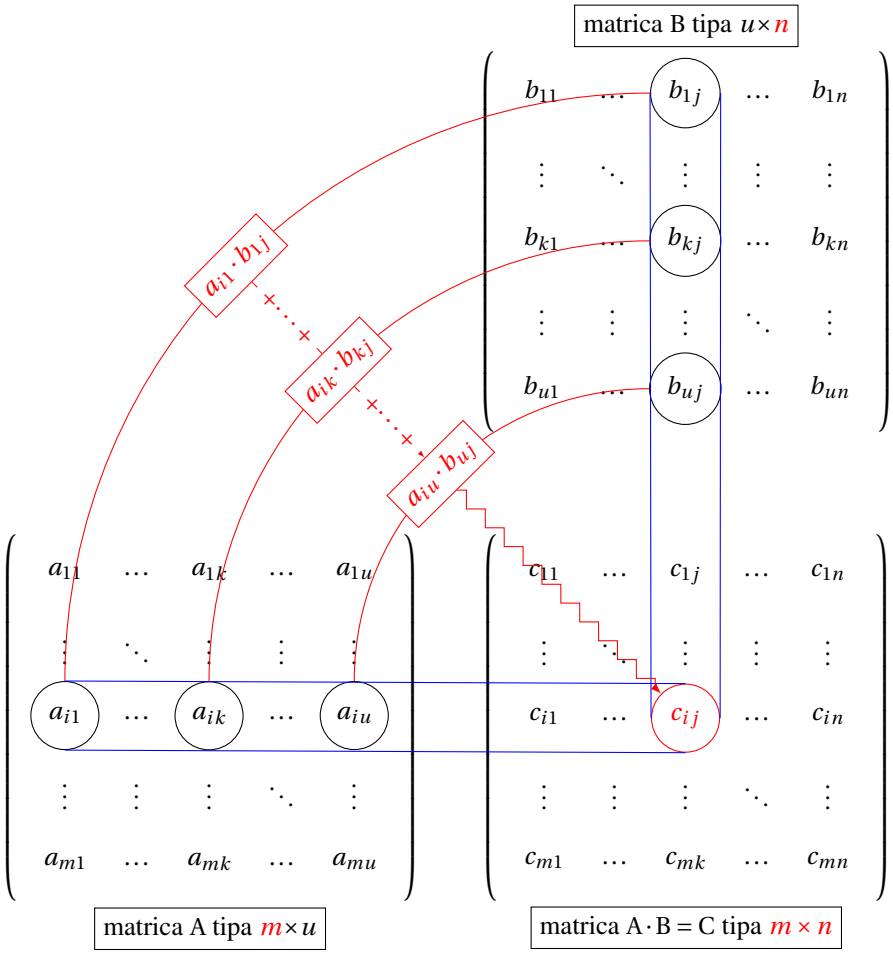
$c_{ij} = 0$

$\forall i \in [1, m]$

  $\forall j \in [1, n]$

    $\forall k \in [1, u]$

      $c_{ij} = c_{ij} + a_{ik} \cdot b_{kj}$



4.1.2 Sustavi linearnih jednadžbi

- Sustav od M linearnih jednadžbi $j = 1 \cdots \mathrm{M}$, s N nepoznanica $x_j,\ j = 1, 2, \ldots, \mathrm{N}$,

$$\begin{array}{ccccccccccc} a_{11} \cdot x_1 & + & a_{12} \cdot x_2 & + & a_{13} \cdot x_3 & + & \cdots & + & a_{1\mathrm{N}} \cdot x_\mathrm{N} & = & b_1 \\ a_{21} \cdot x_1 & + & a_{22} \cdot x_2 & + & a_{23} \cdot x_3 & + & \cdots & + & a_{2\mathrm{N}} \cdot x_\mathrm{N} & = & b_2 \\ a_{31} \cdot x_1 & + & a_{32} \cdot x_2 & + & a_{33} \cdot x_3 & + & \cdots & + & a_{3\mathrm{N}} \cdot x_\mathrm{N} & = & b_3 \\ & & & & & & \cdots & & & & \\ a_{\mathrm{M}1} \cdot x_1 & + & a_{\mathrm{M}2} \cdot x_2 & + & a_{\mathrm{M}3} \cdot x_3 & + & \cdots & + & a_{\mathrm{MN}} \cdot x_\mathrm{N} & = & b_\mathrm{M} \end{array} \tag{10}$$

  možemo zapisati u matričnom oblik

$$\mathrm{A} \cdot \mathrm{X} = \mathrm{B} \tag{11}$$

  gdje je $\mathrm{A} = [a_{ij}]$ matrica sustava, $\mathrm{X} = [x_j]$ matrica nepoznanica te $\mathrm{B} = [b_i]$ matrica slobodnih koeficijenata.

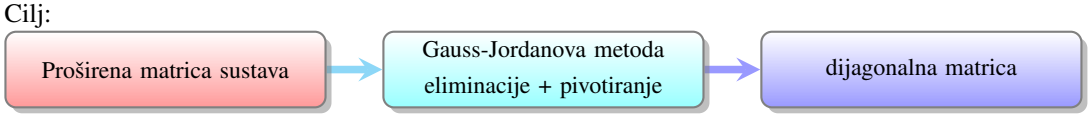
- Radi jednostavnosti često koristimo zapis pomoću proširene matrice sustava

$$\mathrm{A}_p = \mathrm{A}|\mathrm{B} \tag{12}$$

- Sustav linearnih jednadžbi može biti
  - nerješiv;
  - rješiv (Kronecker-Capelli TM) $\Leftrightarrow \mathrm{rang}(\mathrm{A}_p) = \mathrm{rang}(\mathrm{A})$
    - $\mathrm{rang}(\mathrm{A}) = \mathrm{M} \Rightarrow$ sustav je rješiv
    - Cramerov sustav ($\mathrm{rang}(\mathrm{A}) = \mathrm{N} = \mathrm{M}$) ima jedinstveno rješenje (numerički prezahtjevno)

$$x_i = \frac{\det(\mathrm{A}_i)}{\det(\mathrm{A})} \tag{13}$$

      gdje A_i dobijemo tako da i-ti stupac od A zamijenimo s B
    - $\mathrm{rang}(\mathrm{A}) < \mathrm{N} \Rightarrow \infty$ rješenja koja se mogu izraziti preko $\mathrm{N} - \mathrm{rang}(\mathrm{A})$ parametara
- Elementarne transformacije (ne mijenjaju rješenje)
  - zamjene poretka jednadžbi (redaka proširene matrice)
  - množenje jednadžbe (retka) skalarom
  - dodavanje jedne jednadžbe drugoj (retka retku)
- Ostale zamjene:
  - zamjene stupaca j i k odgovaraju zamjeni rješenja $x_j \leftrightarrow x_k$
- Cilj:



- Primjer sustava od 4 linearne jednadžbe s 4 nepoznanice:

$$\begin{aligned} x_2 + 2\,x_3 - x_4 &= 1 \\ x_1 + x_3 + x_4 &= 4 \\ -x_1 + x_2 - x_4 &= 2 \\ 2\,x_2 + 3\,x_3 - x_4 &= 7 \end{aligned}$$

  - matrica sustava A i matrica stupac b

$$\mathrm{A} = \begin{pmatrix} 0 & 1 & 2 & -1 \\ 1 & 0 & 1 & 1 \\ -1 & 1 & 0 & -1 \\ 0 & 2 & 3 & -1 \end{pmatrix}, \qquad b = \begin{pmatrix} 1 \\ 4 \\ 2 \\ 7 \end{pmatrix}.$$

  - proširena matrica sustava

$$\mathrm{A}|b = \left(\begin{array}{cccc|c} 0 & 1 & 2 & -1 & 1 \\ 1 & 0 & 1 & 1 & 4 \\ -1 & 1 & 0 & -1 & 2 \\ 0 & 2 & 3 & -1 & 7 \end{array}\right). \tag{14}$$

  - Svođenje na gornju trokutastu (Gaussova metoda eliminacije)

$$\left(\begin{array}{cccc|c} 0 & 1 & 2 & -1 & 1 \\ 1 & 0 & 1 & 1 & 4 \\ -1 & 1 & 0 & -1 & 2 \\ 0 & 2 & 3 & -1 & 7 \end{array}\right) \xrightarrow{\text{zamjena redaka}} \left(\begin{array}{cccc|c} 1 & 0 & 1 & 1 & 4 \\ -1 & 1 & 0 & -1 & 2 \\ 0 & 1 & 2 & -1 & 1 \\ 0 & 2 & 3 & -1 & 7 \end{array}\right) \xrightarrow{\text{R1+R2}\to\text{R2}}$$

$$\left(\begin{array}{cccc|c} 1 & 0 & 1 & 1 & 4 \\ 0 & 1 & 1 & 0 & 6 \\ 0 & 1 & 2 & -1 & 1 \\ 0 & 2 & 3 & -1 & 7 \end{array}\right) \xrightarrow[\text{R4}-2\text{R2}\to\text{R4}]{\text{R3}-\text{R2}\to\text{R3}} \left(\begin{array}{cccc|c} 1 & 0 & 1 & 1 & 4 \\ 0 & 1 & 1 & 0 & 6 \\ 0 & 0 & 1 & -1 & -5 \\ 0 & 0 & 1 & -1 & -5 \end{array}\right) \xrightarrow{\text{R4}-\text{R3}\to\text{R4}}$$

$$\left(\begin{array}{cccc|c} 1 & 0 & 1 & 1 & 4 \\ 0 & 1 & 1 & 0 & 6 \\ 0 & 0 & 1 & -1 & -5 \\ 0 & 0 & 0 & 0 & 0 \end{array}\right) \xrightarrow[\text{R2}-\text{R3}\to\text{R2}]{\text{R1}-\text{R3}\to\text{R1}} \left(\begin{array}{cccc|c} 1 & 0 & 0 & 2 & 9 \\ 0 & 1 & 0 & 1 & 11 \\ 0 & 0 & 1 & -1 & -5 \\ 0 & 0 & 0 & 0 & 0 \end{array}\right)$$

  - Je li moguće svesti na dijagonalnu matricu (Gauss-Jordanova metoda eliminacije)?
  - Što su rješenja?

4.2 Zadatak

a) Osmislite algoritam koji radi Gauss-Jordanovu metodu eliminacije, odnosno kako napraviti transformaciju sustava (10) u gornje trokutasti pa dijagonalni s jediničnom matricom sustava

$$\begin{array}{l} a_{11}x_1 + a_{12}x_2 + a_{13}x_3 + s + a_{1\mathrm{N}}x_\mathrm{N} = b_1 \\ a_{21}x_1 + a_{22}x_2 + a_{23}x_3 + \cdots + a_{2\mathrm{N}}x_\mathrm{N} = b_2 \\ a_{31}x_1 + a_{32}x_2 + a_{33}x_3 + \cdots + a_{3\mathrm{N}}x_\mathrm{N} = b_3 \\ \cdots \\ a_{\mathrm{M}1}x_1 + a_{\mathrm{M}2}x_2 + a_{\mathrm{M}3}x_3 + \cdots + a_{\mathrm{MN}}x_\mathrm{N} = b_\mathrm{M} \end{array} \quad \Rightarrow \quad \begin{array}{r} g_{11}x_1 + g_{12}x_2 + g_{13}x_3 + \cdots + g_{1\mathrm{N}}x_\mathrm{N} = y_1 \\ g_{22}x_2 + g_{23}x_3 + \cdots + g_{2\mathrm{N}}x_\mathrm{N} = y_2 \\ g_{33}x_3 + \cdots + g_{3\mathrm{N}}x_\mathrm{N} = y_3 \\ \cdots \\ g_{\mathrm{MN}}x_\mathrm{N} = y_\mathrm{M} \end{array} \quad \Rightarrow \quad \begin{array}{r} x_1 = r_1 \\ x_2 = r_2 \\ x_3 = r_3 \\ \cdots \\ x_\mathrm{N} = r_\mathrm{M} \end{array}$$

b) Kako numeričke greške svesti na minimum? Promotrimo neke specifične slučajeve:

- Što ako je na dijagonali 0 kao u prethodnom primjeru u (14)?
- Mogu li nastati numerički problemi ako, nakon „nuliranja" elemenata ispod prvog elementa na dijagonali, dobijemo sustav sličan ovome

$$\begin{pmatrix} 1 & 3 & 4 & 6 \\ 0 & 10^{-8} & 358 & 27 \\ 0 & -91 & 55 & 9 \\ 0 & 7 & 76 & 923 \end{pmatrix} \cdot \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \end{pmatrix}$$

  - Postizanje numerički stabilnijeg proračuna: pivotiranjem (parcijalno - zamjenom redaka, potpuno - zamjena redak i stupaca) na dijagonalu dovodimo najveći preostali element.
  - Dakle, prije „nuliranja" elementa ispod 2. elementa na dijagonali, možemo transformirati sustav zamjenom 2. i 3. retka proširene matrice što odgovara zamjenama istih redaka u matrici sustava A i matrici slobodnih koeficijenata B = $[y_i]$:

$$\begin{pmatrix} 1 & 3 & 4 & 6 \\ 0 & -91 & 55 & 9 \\ 0 & 10^{-8} & 358 & 27 \\ 0 & 7 & 76 & 923 \end{pmatrix} \cdot \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} y_1 \\ y_3 \\ y_2 \\ y_4 \end{pmatrix}$$

    ili zamjenom 2. i 3. stupca proširene matrice što odgovara zamjeni stupaca u A i redaka u rješenju X

$$\begin{pmatrix} 1 & 4 & 3 & 6 \\ 0 & 358 & 10^{-8} & 27 \\ 0 & 55 & -91 & 9 \\ 0 & 76 & 7 & 923 \end{pmatrix} \cdot \begin{pmatrix} x_1 \\ x_3 \\ x_2 \\ x_4 \end{pmatrix} = \begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \end{pmatrix}$$

  - Može li još bolje?

c) Testirajte kod tako da za istu matricu sustava

$$\mathrm{A} = \begin{pmatrix} 4 & -2 & 1 \\ 3 & 6 & -4 \\ 2 & 1 & 8 \end{pmatrix}$$

riješite skup više linearnih jednadžbi,

$$\mathrm{A}\cdot\mathrm{X}_i = \mathrm{B}_i,\ i = 1,2,3$$

$$\begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ b_3 \end{pmatrix}$$

za vektore B

$$\mathrm{B}_1 = \begin{pmatrix} 12 \\ -25 \\ 32 \end{pmatrix},\ \mathrm{B}_2 = \begin{pmatrix} 4 \\ -10 \\ 22 \end{pmatrix},\ \mathrm{B}_3 = \begin{pmatrix} 20 \\ -30 \\ 40 \end{pmatrix}.$$

Rješenje bi trebalo biti

$$\mathrm{X}_1 = \begin{pmatrix} 1 \\ -2 \\ 4 \end{pmatrix},\ \mathrm{X}_2 = \begin{pmatrix} 0.312 \\ -0.038 \\ 2.677 \end{pmatrix},\ \mathrm{X}_3 \begin{pmatrix} 2.319 \\ -2.965 \\ 4.790 \end{pmatrix}.$$

Koraci prema rješenjima:

1. Iskoristite definicije iz Numerical Recipes priložene u arhivi NR_GJ.zip. Raspakirajte arhivu u direktorij bez razmaka u putanji koja nije preduga (npr. u E:). GJ.c sadrži funkciju

```
/* Funkcija koja izvodi Gauss-Jordanovu eliminaciju */
// Ulazni elementi:  ai - sadrzi koeficijente uz nepoznanice = A
//                   n  - dimenzija matrice ai
//                   x  - slobodni koeficijenti = b
//                   m  - broj rjesenja
gaussj(ai,n,x,m);
// Izlazni elementi: ai - inverzna matrica od ulazne
//                   x  - vrijednosti nepoznanica
```

koja Gauss-Jordanovom metodom eliminacije rješava m sustava od n linearnih jednadžbi s n nepoznanica te s istom matricom sustava A,

$$\mathrm{A}\cdot\mathrm{X}_i = \mathrm{B}_i, \quad i = 1\cdots m\,. \tag{15}$$

Definicije dodatnih funkcije (matrix, ivector, nrerror, ...) potrebnih njoj nalaze se u nrutil.h i nrutil.c. Funkcija matrix rezervira mjesto u memoriju za npr. matricu A tipa $r \times s$

```
float **a;
a = matrix(1, r, 1, s);
```

pa u $a[i][j]$ pohranjujemo element u $1 \le i \le r$ redku i $1 \le j \le s$ stupcu. Veza pokazivača i polja te primjer rezervacije memorije dani su u dodatku A.

2. Razmislite može li se Gauss-Jordanova metoda iskoristit za proračun inverzne matrice $\mathrm{A}^{-1} = [ai_{ij}]$. Podsjetnik: $\mathrm{A} \cdot \mathrm{A}^{-1} = \mathrm{I}$.

3. Doradite start_GJ.c

   - Učitajte broj broj sustava jednadžbi m i red matrice sustava n iz datoteke jednadzbe.txt (u dodatku B dan je primjer pisanja i čitanja u i iz datoteka).
   - Koristeći matrix kreirajte dvodimenzionalna polja, odnosno matrice a i ai tipa $n \times n$ te x i b matrice tipa $n \times m$.
   - Inicijalizirajte polja a i b s elemntima zapisanim u datoteci jednadzbe.txt.
   - Kopirajte polje a u polje ai te polje b u polje x.
   - Pozovite gasussj.
   - Rješenja i inverznu matricu pohranite u datoteku rjesenja.txt.

4. Obavite provjeru rješenja. Definirajte nova potrebna polja koristeći matrix. Ispitajte

$$\mathrm{A} \cdot \mathrm{A}^{-1} = \mathrm{I} \tag{16}$$

$$\mathrm{A} \cdot \mathrm{X}_i = \mathrm{B}_i \tag{17}$$

5. Upišite u datoteku jednadzbeC.txt podatke iz zadatka pod c) na prethodnoj stranici. Pokrenite program. Usporedite dobivena i dana rješenja.

Opširnije:

- Odgovori na predavanjima 30.05. i 3.6.
- Dodatna literatura je na moodle-u.

A Pokazivači

.

```c
#include <stdio.h>
#include <stdlib.h>

int main () {
   int niz[5] = {10,20,30,40,50};
   int *p, *novi;
   int i;

   // elementi niza
   for ( i = 0; i < 5; i++ ) {
      printf("%d, ", niz[i]);
   }
   printf("\n");

   // ispis pomocu pokazivaca
   p = niz;
   for ( i = 0; i < 5; i++ ) {
      printf("%d, ", *(p + i) );
   }
   printf("\n");

   // ispis pomocu pokazivaca
   p = &niz[2];
   for ( i = 0; i < 5; i++ ) {
      printf("%d, ", *(p - 2 + i) );
   }
   printf("\n");

   // rezervira memoriju za novi niz
   novi = malloc(3 * sizeof(int));
   novi[0] = 11;
   novi[1] = 22;
   novi[2] = 33;
   for (i = 0; i < 3; i++) {
     printf("%d, ", novi[i]);
   }
   printf("\n");
   // brise novi niz iz memorije
   free(novi);

   return 0;
}
```

B Datoteke

.

```c
#include <stdio.h>

main() {

  FILE *dat;
  char tekst[125];
  int iw=1, jw=5, ir, jr;

  // otvara-kreira datoteku za pisanje
  dat = fopen("test.txt", "w");
  // upis teksta i brojeva
  fprintf(dat, "2 cijela broja:\n");
  fprintf(dat, "%d  %d",iw,jw);
  // zatvaranje datoteke
  fclose(dat);

  // otvara datoteku za citanje
  dat = fopen("test.txt", "r");
  // prelazak u novi red oznacava kraj podatka kojeg cita
  fgets(tekst, 125, (FILE*)dat);
  // razmak oznacava kraj podatka kojeg cita
  fscanf(dat, "%d %d", &ir, &jr);
  // zatvaranje datoteke
  fclose(dat);

  printf("TEST: %d = %d,  %d = %d\n", iw, ir, jw, jr);
  return 0;
}
```