Tiesinių lygčių sistemos

II praktikos užduotis

Šiai užduočiai reikia sugeneruoti lygties $AX = B$ duomenis pačiam. Jei užduotyje nenurodyta kitaip, naudokite tokį būdą, kad gautumėte paprastus sprendinius: Sugeneruokite dydžio $N \times N$ matricą A. Pasirinkite nesudėtingą N ilgio vektorių–sprendinį X_{spr}, pvz. vien tik vienetus, ar 01 seką ir pan. Sudauginę AX, gaunate vektorių B. Dabar X_{spr} "pamirštate" ir ieškote nurodytais būdais lygties $AX = B$ sprendinių.

Realizuojami metodai turi būti suprogramuoti jūsų pačių, negalima naudoti *built-in* funckcijų ar bibliotekų tiesinei algebrai (nebent atsakymui pasitikrinti). Stenkitės optimizuoti naudodami vektorines operacijas ir išvengdami ciklų, kur nėra būtina.

Užduotis pasiimkite iš lentelės:

Nr.	Užduotis	Nr.	Užduotis	Nr.	Užduotis	Nr.	Užduotis
1	1, 11	15	5, 13	29	7, 16	43	1, 14
2	2, 12	16	6, 14	30	8, 11	44	2, 15
3	3, 13	17	7, 15	31	1, 15	43	9, 16
4	4, 14	18	8, 11	32	2, 16	44	10, 11
5	5, 15	19	9, 12	33	9, 12	45	1, 13
6	6, 16	20	10, 13	34	10, 13	46	2, 14
7	7, 11	21	1, 14	35	1, 12	47	3, 15
8	8, 12	22	2, 15	36	2, 13	48	4, 16
9	9, 13	23	1, 16	37	3, 14	49	5, 11
10	10, 14	24	2, 11	38	4, 15	50	6, 12
11	1, 15	25	3, 12	39	5, 16	51	7, 13
12	2, 16	26	4, 13	40	6, 11	52	8, 14
13	3, 11	27	5, 14	41	7, 12		
14	4, 12	28	6, 15	42	8, 13		

Užduočių sąrašas:

1. Išspręskite lygtį naudodami Gauso metodą su pagrindinio elemento parinkimu iš tvarkomo stulpelio elementų. Naudokite pilną N dydžio matricą. Apskaičiuokite algoritmo darbo laiką keisdami N ir panaudokite tai įvertinti algoritmo sudėtingumą duomenų dydžio atžvilgiu.

2. Išspręskite lygtį naudodami Gauso metodą su pagrindinio elemento parinkimo iš tvarkomos eilutės elementų. Naudokite pilną N dydžio matricą. Apskaičiuokite algoritmo darbo laiką keisdami N ir panaudokite tai įvertinti algoritmo sudėtingumą duomenų dydžio atžvilgiu.

3. Išspręskite netiesinę lygčių sistemą $AX = F(X)$ tikslumu ε naudodami itericijų metodą $AX^{(k+1)} = F(X^{(k)})$ ir paklaidą skaičiuodami su maksimumo norma. Kiekvienoje iteracijoje gaunamą lygčių sistemą spręskite panaudodami LU dekompoziciją. Susikurkite matricą A ir funkciją F(X) tokiu būdu: Sugeneruokite dydžio $N \times N$ matricą A iš -1, 0 ir 1. Perskaičiuokite pagrindinės įstrižainės elementus $a_{ii} = \sum_{j=1}^{N} |a_{ij}|$. Pasirinkite N ilgio vektorių–sprendinį X_{spr}, sudarytą tik iš vienetų. Suskaičiuokite vektorių $Y = AX$ ir apsibrėžkite tokią funkciją $F(X) = (X - X_{spr})^2 - Y$. Dabar "pamirškite" sprendinį X_{spr} ir spręskite susikonstruotą netiesinę lygtį $AX = F(X)$.

Apskaičiuokite atskirai a)LU dekompozicijos algoritmo darbo laiką keisdami N ir b)pačių lygčių su trikampėmis matricomis sprendimo laiką. Panaudokite tai įvertinti šių algoritmų sudėtingumą duomenų dydžio atžvilgiu.

4. Išspręskite netiesinę lygčių sistemą $AX = F(X)$ tikslumu ε naudodami itericijų metodą $AX^{(k+1)} = F(X^{(k)})$ ir paklaidą skaičiuodami su maksimumo norma. Kiekvienoje iteracijoje gaunamą lygčių sistemą spręskite panaudodami Choleskio dekompoziciją. Susikurkite matricą A ir funkciją F(X) tokiu būdu: Sugeneruokite **simetrinę** dydžio $N \times N$ matricą A iš -1, 0 ir 1. Perskaičiuokite pagrindinės įstrižainės elementus $a_{ii} = \sum_{j=1}^{N} |a_{ij}|$. Pasirinkite N ilgio vektorių–sprendinį X_{spr}, sudarytą tik iš vienetų. Suskaičiuokite vektorių $Y = AX$ ir apsibrėžkite tokią funkciją $F(X) = (X - X_{spr})^2 - Y$. Dabar "pamirškite" sprendinį X_{spr} ir spręskite susikonstruotą netiesinę lygtį $AX = F(X)$.

   Apskaičiuokite atskirai a)Choleskio algoritmo darbo laiką keisdami N ir b)pačių lygčių su trikampėmis matricomis sprendimo laiką. Panaudokite tai įvertinti šių algoritmų sudėtingumą duomenų dydžio atžvilgiu.

5. Išspręskite netiesinę lygčių sistemą $AX = F(X)$ tikslumu ε naudodami itericijų metodą $AX^{(k+1)} = F(X^{(k)})$ ir paklaidą skaičiuodami su maksimumo norma. Kiekvienoje iteracijoje gaunama lygčių sistemą spręskite panaudodami LU dekompoziciją. Trijsžainę matricą $A_{N\times N}$ ir N ilgio vektorinė f-ją $F(X)$ susikurkite tokiu būdu:

$$A = \begin{pmatrix} 2 & -1 & 0 & 0 & \cdots & 0 & 0 \\ -1 & 2 & -1 & 0 & \cdots & 0 & 0 \\ 0 & -1 & 2 & -1 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & -1 & 2 \end{pmatrix}, \quad F(X) = \begin{pmatrix} f_1 \\ f_2 \\ \cdots \\ f_i \\ \cdots \\ f_N \end{pmatrix} = \begin{pmatrix} c\,(x_1^2+1) \\ c\,(x_2^2+1) \\ \cdots \\ c\,(x_i^2+1) \\ \cdots \\ c\,(x_N^2+1) \end{pmatrix}, \quad c = \frac{1}{(N+1)^2}$$

   Apskaičiuokite atskirai a)LU dekompozicijos algoritmo darbo laiką keisdami N ir b)pačių lygčių su trikampėmis matricomis sprendimo laiką. Panaudokite tai įvertinti šių algoritmų sudėtingumą duomenų dydžio atžvilgiu.

   Pastaba LU dekompozicija nėra efektyvus būdas spręsti tokiam uždaviniui, nes daugelis A elementų 0, bet tikslas išmėginti metodą.

6. Išspręskite netiesinę lygčių sistemą $AX = F(X)$ tikslumu ε naudodami itericijų metodą $AX^{(k+1)} = F(X^{(k)})$ ir paklaidą skaičiuodami su maksimumo norma. Kiekvienoje iteracijoje gaunama lygčių sistemą spręskite panaudodami Choleskio dekompoziciją. Trijįstrižainę matricą $A_{N\times N}$ ir N ilgio vektorinė f-ją $F(X)$ susikurkite tokiu būdu:

$$A = \begin{pmatrix} 2 & -1 & 0 & 0 & \cdots & 0 & 0 \\ -1 & 2 & -1 & 0 & \cdots & 0 & 0 \\ 0 & -1 & 2 & -1 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & -1 & 2 \end{pmatrix}, \quad F(X) = \begin{pmatrix} f_1 \\ f_2 \\ \cdots \\ f_i \\ \cdots \\ f_N \end{pmatrix} = \begin{pmatrix} c\,(x_1^2+1) \\ c\,(x_2^2+1) \\ \cdots \\ c\,(x_i^2+1) \\ \cdots \\ c\,(x_N^2+1) \end{pmatrix}, \quad c = \frac{1}{(N+1)^2}$$

   Apskaičiuokite atskirai a)Choleskio algoritmo darbo laiką keisdami N ir b)pačių lygčių su trikampėmis matricomis sprendimo laiką. Panaudokite tai įvertinti šių algoritmų sudėtingumą duomenų dydžio atžvilgiu.

Pastaba Choleskio dekompozicija nėra efektyvus būdas spręsti tokiam uždaviniui, nes daugelis A elementų 0, bet tikslas išmėginti metodą.

7. Išspręskite netiesinę lygčių sistemą $AX = F(X)$ tikslumu ε naudodami itericijų metodą $AX^{(k+1)} = F(X^{(k)})$ ir paklaidą skaičiuodami su maksimumo norma. Kiekvienoje iteracijoje gaunama lygčių sistemą spręskite panaudodami LU dekompoziciją. Penkiaįstrižainę matricą $A_{N\times N}$ ir N ilgio vektorinė f-ją $F(X)$ susikurkite tokiu būdu:

$$A = \begin{pmatrix} 30 & -16 & 1 & 0 & 0 & \cdots & 0 & 0 & 0 \\ -16 & 30 & -16 & 1 & 0 & \cdots & 0 & 0 & 0 \\ 1 & -16 & 30 & -16 & 1 & \cdots & 0 & 0 & 0 \\ 0 & 1 & -16 & 30 & -16 & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & 0 & \cdots & 1 & -16 & 30 \end{pmatrix}, \quad F(X) = \begin{pmatrix} f_1 \\ f_2 \\ \cdots \\ f_i \\ \cdots \\ f_{N-1} \\ f_N \end{pmatrix} = \begin{pmatrix} c + 2\left(x_2 - 0\right)^2 \\ c + 2\left(x_3 - x_1\right)^2 \\ \cdots \\ c + 2\left(x_{i+1} - x_{i-1}\right)^2 \\ \cdots \\ c + 2\left(x_N - x_{N-2}\right)^2 \\ c + 2\left(0 - x_{N-1}\right)^2 \end{pmatrix},$$

kur $c = \frac{1}{(N+1)^2}$

Apskaičiuokite atskirai a)LU dekompozicijos algoritmo darbo laiką keisdami N ir b)pačių lygčių su trikampėmis matricomis sprendimo laiką. Panaudokite tai įvertinti šių algoritmų sudėtingumą duomenų dydžio atžvilgiu.

Pastaba LU dekompozicija nėra efektyvus būdas spręsti tokiam uždaviniui, nes daugelis A elementų 0, bet tikslas išmėginti metodą.

8. Išspręskite netiesinę lygčių sistemą $AX = F(X)$ tikslumu ε naudodami itericijų metodą $AX^{(k+1)} = F(X^{(k)})$ ir paklaidą skaičiuodami su maksimumo norma. Kiekvienoje iteracijoje gaunama lygčių sistemą spręskite panaudodami Choleskio dekompoziciją. Penkiaįstrižainę matricą $A_{N\times N}$ ir N ilgio vektorinė f-ją $F(X)$ susikurkite tokiu būdu:

$$A = \begin{pmatrix} 30 & -16 & 1 & 0 & 0 & \cdots & 0 & 0 & 0 \\ -16 & 30 & -16 & 1 & 0 & \cdots & 0 & 0 & 0 \\ 1 & -16 & 30 & -16 & 1 & \cdots & 0 & 0 & 0 \\ 0 & 1 & -16 & 30 & -16 & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & 0 & \cdots & 1 & -16 & 30 \end{pmatrix}, \quad F(X) = \begin{pmatrix} f_1 \\ f_2 \\ \cdots \\ f_i \\ \cdots \\ f_{N-1} \\ f_N \end{pmatrix} = \begin{pmatrix} c + 2\left(x_2 - 0\right)^2 \\ c + 2\left(x_3 - x_1\right)^2 \\ \cdots \\ c + 2\left(x_{i+1} - x_{i-1}\right)^2 \\ \cdots \\ c + 2\left(x_N - x_{N-2}\right)^2 \\ c + 2\left(0 - x_{N-1}\right)^2 \end{pmatrix},$$

kur $c = \frac{1}{(N+1)^2}$

Apskaičiuokite atskirai a)Choleskio algoritmo darbo laiką keisdami N ir b)pačių lygčių su trikampėmis matricomis sprendimo laiką. Panaudokite tai įvertinti šių algoritmų sudėtingumą duomenų dydžio atžvilgiu.

Pastaba Choleskio dekompozicija nėra efektyvus būdas spręsti tokiam uždaviniui, nes daugelis A elementų 0, bet tikslas išmėginti metodą.

9. Išspręskite netiesinę lygčių sistemą $AX = F(X)$ tikslumu ε naudodami itericijų metodą $AX^{(k+1)} = F(X^{(k)})$ ir paklaidą skaičiuodami su maksimumo norma. Kiekvienoje iteracijoje gaunama lygčių sistemą spręskite panaudodami LU dekompoziciją. Trijsžainę matricą $A_{N\times N}$ ir N ilgio vektorinė

f-ją $F(X)$ susikurkite tokiu būdu:

$$A = \begin{pmatrix} 2 & -1 & 0 & 0 & \cdots & 0 & 0 \\ -1 & 2 & -1 & 0 & \cdots & 0 & 0 \\ 0 & -1 & 2 & -1 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & 2 & -1 \\ 0 & 0 & 0 & 0 & \cdots & -1 & 2 \end{pmatrix}, \quad F(X) = \begin{pmatrix} f_1 \\ f_2 \\ \cdots \\ f_i \\ \cdots \\ f_{N-1} \\ f_N \end{pmatrix} = \begin{pmatrix} c + 2\left(x_2 - 0\right)^2 \\ c + 2\left(x_3 - x_1\right)^2 \\ \cdots \\ c + 2\left(x_{i+1} - x_{i-1}\right)^2 \\ \cdots \\ c + 2\left(x_N - x_{N-2}\right)^2 \\ c + 2\left(0 - x_{N-1}\right)^2 \end{pmatrix},$$

kur $c = \frac{1}{(N+1)^2}$

Apskaičiuokite atskirai a)LU dekompozicijos algoritmo darbo laiką keisdami N ir b)pačių lygčių su trikampėmis matricomis sprendimo laiką. Panaudokite tai įvertinti šių algoritmų sudėtingumą duomenų dydžio atžvilgiu.

Pastaba LU dekompozicija nėra efektyvus būdas spręsti tokiam uždaviniui, nes daugelis A elementų 0, bet tikslas išmėginti metodą.

10. Išspręskite netiesinę lygčių sistemą $AX = F(X)$ tikslumu ε naudodami itericijų metodą $AX^{(k+1)} = F(X^{(k)})$ ir paklaidą skaičiuodami su maksimumo norma. Kiekvienoje iteracijoje gaunama lygčių sistemą spręskite panaudodami Choleskio dekompoziciją. Trijšžainę matricą $A_{N\times N}$ ir N ilgio vektorinė f-ją $F(X)$ susikurkite tokiu būdu:

$$A = \begin{pmatrix} 2 & -1 & 0 & 0 & \cdots & 0 & 0 \\ -1 & 2 & -1 & 0 & \cdots & 0 & 0 \\ 0 & -1 & 2 & -1 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & 2 & -1 \\ 0 & 0 & 0 & 0 & \cdots & -1 & 2 \end{pmatrix}, \quad F(X) = \begin{pmatrix} f_1 \\ f_2 \\ \cdots \\ f_i \\ \cdots \\ f_{N-1} \\ f_N \end{pmatrix} = \begin{pmatrix} c + 2\left(x_2 - 0\right)^2 \\ c + 2\left(x_3 - x_1\right)^2 \\ \cdots \\ c + 2\left(x_{i+1} - x_{i-1}\right)^2 \\ \cdots \\ c + 2\left(x_N - x_{N-2}\right)^2 \\ c + 2\left(0 - x_{N-1}\right)^2 \end{pmatrix},$$

kur $c = \frac{1}{(N+1)^2}$

Apskaičiuokite atskirai a)Choleskio dekompozicijos algoritmo darbo laiką keisdami N ir b)pačių lygčių su trikampėmis matricomis sprendimo laiką. Panaudokite tai įvertinti šių algoritmų sudėtingumą duomenų dydžio atžvilgiu.

Pastaba Choleskio dekompozicija nėra efektyvus būdas spręsti tokiam uždaviniui, nes daugelis A elementų 0, bet tikslas išmėginti metodą.

11. Išspręskite lygtį naudodami Zeidelio metodą. Matrica A - tokia, kad metodas konverguotų maksimumo normoje (reikia sumuoti eilutės elementus). Paklaidą skaičiuokite naudodami maksimumo L_∞ normą su tikslumu ε. Palyginkite su veikimo laiku sprendžiant tiksliuoju metodu iš anksktesnio uždavinio au tuo pačiu N.

12. Išspręskite lygtį naudodami Jakobio metodą. Matrica A - tokia, kad metodas konverguotų maksimumo normoje (reikia sumuoti eilutės elementus). Paklaidą skaičiuokite naudodami L_∞ maksimumo normą su tikslumu ε. Palyginkite su veikimo laiku sprendžiant tiksliuoju metodu iš anksktesnio uždavinio au tuo pačiu N.

13. Išspręskite lygtį naudodami gradientų (didžiausio nuolydžio) metodą. Susikurkite simetrinę teigiamai apibrėžtą matricą A tokiu būdu: Sugeneruokite **simetrinę** dydžio $N \times N$ matricą A.

Perskaičiuokite pagrindinės įstrižainės elementus $a_{ii} = \sum_{j=1}^{N} |a_{ij}|$ (tai garantuos teigiamą apibrėžtumą). Paklaidą skaičiuokite naudodami maksimumo L_∞ normą su tikslumu ε. Palyginkite su veikimo laiku sprendžiant tiksliuoju metodu iš ansktesnio uždavinio au tuo pačiu N.

14. Išspręskite lygtį naudodami Zeidelio metodą. Matrica A - tokia, kad metodas konverguotų integralinėje normoje (reikia sumuoti stulpelio elementus padalintus iš i-ojo pagrindinės diagonalės elemento). Paklaidą skaičiuokite naudodami integralinę (sumavimo) l_1 normą su tikslumu ε. Palyginkite su veikimo laiku sprendžiant tiksliuoju metodu iš ansktesnio uždavinio au tuo pačiu N.

15. Išspręskite lygtį naudodami Jakobio metodą. Matrica A - tokia, kad metodas konverguotų integralinėje normoje (reikia sumuoti stulpelio elementus padalintus iš i-ojo pagrindinės diagonalės elemento). Paklaidą skaičiuokite naudodami integralinę (sumavimo) l_1 normą su tikslumu ε. Palyginkite su veikimo laiku sprendžiant tiksliuoju metodu iš ansktesnio uždavinio au tuo pačiu N.

16. Išspręskite lygtį naudodami gradientų (didžiausio nuolydžio) metodą. Susikurkite simetrinę teigiamai apibrėžtą matricą A tokiu būdu: Sugeneruokite **simetrinę** dydžio $N \times N$ matricą A. Perskaičiuokite pagrindinės įstrižainės elementus $a_{ii} = \sum_{j=1}^{N} |a_{ij}|$ (tai garantuos teigiamą apibrėžtumą). Paklaidą skaičiuokite naudodami integralinę (sumavimo) l_1 normą su tikslumu ε. Palyginkite su veikimo laiku sprendžiant tiksliuoju metodu iš ansktesnio uždavinio au tuo pačiu N.

Jums gali praversti šios MATLAB funkcijos ir patarimai:

rand - generuoja atsitiktinai matricą.
diag - leidžia matricoms prisikirti reikšmes pagal diagonalės elementus.
chol - atlikus operaciją *[R,p]=chol(A)*, gauname: R - choleckio dekompozija, $p = 0$ jei matrica teigiamai apibrėžta (tai efektyvus būdas patikrinti teigiamą apibrėžtumą).
lu - LU dekompozicija.
$A \setminus B$ - išsprendžia lygtį $AX = B$.
MATLAB yra vektorinė programavimo kalba - norint efektyviai skaičiuoti, pvz. skaliarinę sandaugą $X \cdot Y$, reikia rašyti ne ciklą, o naudoti komandą $X * Y'$ kai abu vektoriai - eilutės. Panaudokite tai, stipriai pagreitės veikimas.
sum - sumuoja vektoriaus komponentes.
max - didžiausia vektoriaus komponentė.