Blatt 3: LR-Zerlegung **MNEU**

(Theorieteil)

Aufgabe 1 :

(a) Berechnen Sie per Hand die LR-Zerlegung der Matrix

$$A = \begin{pmatrix} 1 & 2 & 3 \\ 1 & 0 & 1 \\ 2 & 0 & 0 \end{pmatrix}.$$

(b) Lösen Sie mit der Matrix aus Aufgabe (a) das LGS

$$A\,x = y$$

mit $y = (1,1,1)^T$ unter Verwednung der Methode der LR-Zerlegung.

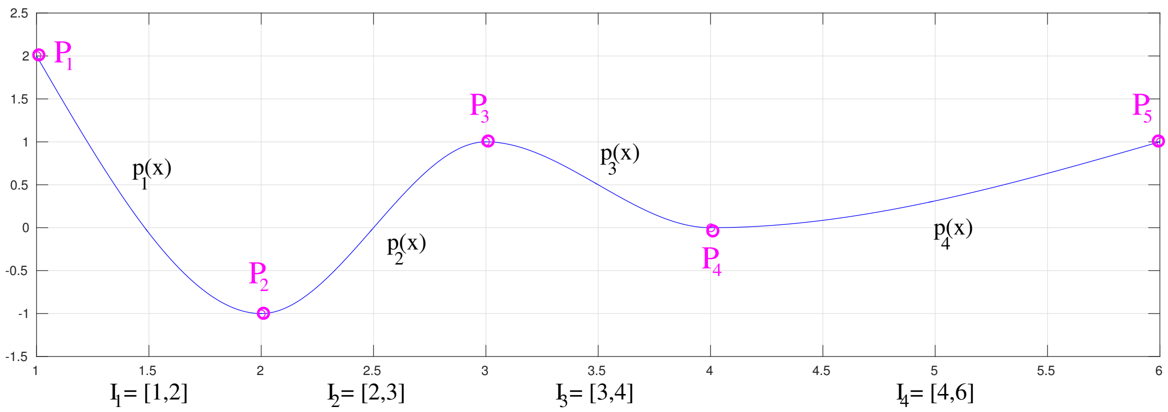
(c) Prüfen Sie Ihr Ergebnis

Aufgabe 2 (kubische C^2-Splines):

(a) Lesen Sie sich folgende Beschreibung durch: Ein kubischer C^2-Spline ist eine Funktion $f : I \to \mathbb{R}$, die aus glatt zusammengesetzten kubischen Polynomen besteht:

$$f(x) = \begin{cases} p_1(x) & \text{für} \quad x \in [x_1, x_2] \\ p_2(x) & \text{für} \quad x \in [x_2, x_3] \\ \vdots & \\ p_N(x) & \text{für} \quad x \in [x_{N-1}, x_N] \end{cases} \in C^2(I)\,,$$

mit $I = \cup_{k=1}^{N-1} [x_k, x_{k+1}]$ und $p_j \in \mathbb{P}_3$. Die Position der Übergänge $(x_i, f(x_i))$ ist durch eine Menge von Punkten $P_i = (x_i, y_i = f(x_i))$ vorgegeben:



Glatt zusammengesetzt bedeutet dabei, dass die Polynome nicht nur stetig, sondern auch ein und zwei Mal differenzierbar sein müssen bei den Übergängen. Insgesamt gilt dann für $f \in C^2(I)$:

An den inneren Knoten $j = 2, \ldots, N-1$ gelte

$$\begin{aligned} p_{j-1}(x_j) &= y_j \\ p_j(x_j) &= y_j \\ p_{j-1}^{(k)}(x_j) &= p_j^{(k)}(x_j) && \text{für } k \in \{1,2\} \end{aligned}$$

und an den Randknoten

$$\begin{aligned} p_1(x_1) &= y_1 \\ p_1^{(2)}(x_1) &= 0 \\ p_{N-1}(x_N) &= y_N \\ p_{N-1}^{(2)}(x_N) &= 0 \end{aligned}$$

Das sind $4\,(N-1)$ Gleichungen, die genauso viele DOFs (dof=degree of freedom) also Unbekannte beinhalten. Das führt also auf ein $4\,(N-1) \times 4\,(N-1)$ Gleichungssystem.

(b) **JETZT DIE AUFGABE** Stellen Sie ein solches Gleichungssystem für die Punkte

$$P_1 = (1,2)\,,\ P_2 = (3,-1) \text{ und } P_3 = (4,0)$$

auf.

Aufgabe 3 (sparse matrix / dünn besetzte Matrix):

Gegeben sei die Matrix

$$\mathcal{A} = \begin{pmatrix} 1 & 2 & 0 & 0 & 1 \\ 0 & -1 & 0 & 1 & 2 \\ 2 & 3 & 0 & 4 & 0 \\ 2 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 1 & 0 \end{pmatrix}$$

gespeichert mit Bandbreite $Bb = 3$ im Feld

$$\mathtt{A} = [1\,,\ 2\,,\ 1\,|\,-1\,,\ 1\,,\ 2\,|\,2\,,\ 3\,,\ 4\,|\,2\,,\ 1\,,\ 1\,|\,1\,,\ 1\,,\ 1]$$

Wie lautet das Zeigerfeld $Z \in \mathbb{N}^{3 \cdot Bb}$, das die Einträge den entsprechende Positionen in A zuordnet?

(Praxisteil)

Aufgabe 4 :

Berechnen Sie die Lösung aus Aufgabe 1 mit dem Programm `LGS/MyLGS.m` und der zugehörigen Funktion `LGS/MyLR.m`.

Aufgabe 5 :

Berechnen Sie mit dem Programm `MyLGS.m` und der zugehörigen Funktion `LGS/MyLR.m` die Lösung des LGS

$$\begin{aligned} a + 2\,b + c + d &= 1 \\ 2\,a + b - 2\,c + d &= 2 \\ a - 2\,b - c + d &= 3 \\ a + b + c + 2\,d - 4 &= 0 \end{aligned}$$

Aufgabe 6 :

Berechnen Sie die Lösung aus Aufgabe 2 unter Verwendung des Matlab-Skripts

```
x = A\y;
```

zur Lösung des LGS $Ax = y$. Stellen Sie den berechneten Spline graphisch dar.

Aufgabe 7 (sparse matrix / dünn besetzte Matrix):

(a) Schreiben Sie eine Funktion `y=SparseMatMult(AS,ZS,Bb,x)`, welches die Matrix-Vektor-Multiplikation $y = A\,x$ der Matrix `A` aus Aufgabe 3 unter Ausnutzung der speziellen Datenstruktur durchführt. Verwenden Sie dazu den Vektor

$$x = (1, 2, 3, 4, 5)\,.$$

(b) Schreiben Sie eine Funktion `[AS ZS Bb]=SparseMat(A)`, dass bei einer gegebenen Matrix A die Bandbreite ermittelt und entsprechend sparsam in `AS` abspeichert. Wenden Sie die Funktion zum testen auf die Matrix aus Teil (a) an.

(c) Schreiben Sie eine Funktion `y=MatMult(A,x)`, die eine "ganz normale" Matrix-Vektor-Multiplikation mit einer $N \times N$-Matrix A und einem Vektor $x \in \mathbb{R}^N$ ausführt. Vergleichen Sie die Ergebnisse

```
ys = SparseMatMult(AS,ZS,Bb,x);
yn = MatMult(A,x);
```

in der Maximumnorm `norm(ys-yn,inf)` Sie sollten das gleiche Ergebnis erhalten.

(d) Schreiben Sie die Subfunktion

```
function A = BandRandMat(DIM,Bb)

A = zeros(DIM);

for i=1:DIM
    for jj=1:Bb
        j = randi(DIM);
        while abs(A(i,j))>0 j = randi(DIM); end
        A(i,j)=rand(1)+1;
    end
end
end
```

Sie erhalten eine DIM x DIM-Matrix mit Bandbreite Bb und zufällig gewählten Komponenten, in die Zufallswerte aus $(1, 2)$ gespeichert sind. Verwenden Sie ab jetzt diese Matrix für Ihre Rechnungen. Sie können nun bequem per Parameter `DIM` und `Bb` verschiedene Situationen erzeugen.

(e) Schreiben Sie ein Programm `MyLGS_Blatt3_Sparse.m` mit folgendem Inhalt:

```
DIM = 100;
Bb = 3;

A = BandRandMat(DIM,Bb);
[AS ZS Bb] = SparseMat(A);
x  = ones(DIM,1);

tic; ys = SparseMatMult(AS,ZS,Bb,x); toc;
tic; ym = MatMult(A,x); toc;
```

Vergleichen Sie die Rechenzeiten und spielen Sie mit dem Parameter `DIM`. (Auf jeden Fall bis DIM=10000) Was beobachten Sie?

Lösung 1:

(a) LR-Zerlegung:

$$A = L \cdot R = \begin{pmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 2 & 2 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & 2 & 3 \\ 0 & -2 & -2 \\ 0 & 0 & -2 \end{pmatrix}$$

(b) Zwischenschritt:

$$L\,y = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} \quad \Leftrightarrow \quad y = \begin{pmatrix} 1 \\ 0 \\ -1 \end{pmatrix}$$

Rekursion:

$$R\,x = \begin{pmatrix} 1 \\ 0 \\ -1 \end{pmatrix} \quad \Leftrightarrow \quad x = \frac{1}{2}\begin{pmatrix} 1 \\ -1 \\ 1 \end{pmatrix}$$

(c) Test:

$$\begin{pmatrix} 1 & 2 & 3 \\ 1 & 0 & 1 \\ 2 & 0 & 0 \end{pmatrix} \begin{pmatrix} 1 \\ -1 \\ 1 \end{pmatrix} \frac{1}{2} = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} \qquad \checkmark$$

Lösung 2:

```
A =
    18     2     0     0   -18    -2     0     0
    27     9     3     1     0     0     0     0
    27     6     1     0   -27    -6    -1     0
     1     1     1     1     0     0     0     0
     6     2     0     0     0     0     0     0
     0     0     0     0    24     2     0     0
     0     0     0     0    27     9     3     1
     0     0     0     0    64    16     4     1

y =
     0    -1     0     2     0     0    -1     0
```

(Hier wurden die Zeilen in der Matrix so vertauscht, dass eine LR-Zerlegung problemlos möglich ist)

Lösung 3:

$$\mathtt{Z} = [1,2,5 \mid 2,4,5 \mid 1,2,4 \mid 1,3,5 \mid 2,3,4]$$

Lösung 4:

(...)

Lösung 5:

Das LGS in Matrix-Vektorform lautet:

$$\begin{pmatrix} 1 & 2 & 1 & 1 \\ 2 & 1 & -2 & 1 \\ 1 & -2 & -1 & 1 \\ 1 & 1 & 1 & 2 \end{pmatrix} \begin{pmatrix} a \\ b \\ c \\ d \end{pmatrix} = \begin{pmatrix} 1 \\ 2 \\ 3 \\ 4 \end{pmatrix}$$

Eingabe im Programm:

```
A=[[1 2 1 1];[2 1 -2 1];[1 -2 -1 1];[1 1 1 2]];
y=[1;2;3;4]';

x  = MyLR(A,y);
```

liefert die Lösung

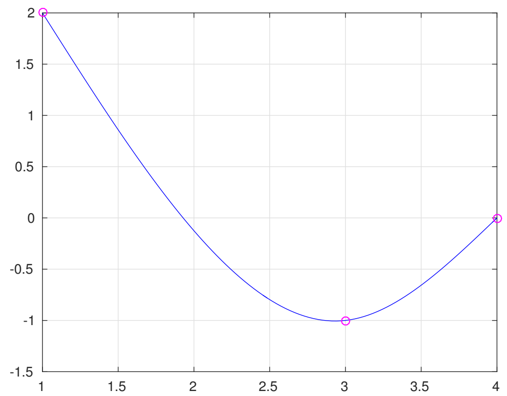
```
x =

   -0.7500    -0.2500    -0.5000     2.7500
```

oder schöner geschrieben:

$$x = \frac{-1}{4} \begin{pmatrix} 3 \\ 1 \\ 2 \\ -11 \end{pmatrix}$$

Lösung 6:

Den Programmcode finden Sie unter LGS/Blatt3/MyLGS_B3A2b.m Der wesentliche Bestandteil der Aufgabe ist es, die Ergebniskurve korrekt durch einen Plot darzustellen. Ihr Plot sollte dann so aussehen:



Lösung 7:

(a) LGS/Blatt3/MyLGS_Blatt3_Aufgabe7a.m

(b) LGS/Blatt3/MyLGS_Blatt3_Aufgabe7b.m

(c) LGS/Blatt3/MyLGS_Blatt3_Aufgabe7c.m

(d) (...)

(e) LGS/Blatt3/MyLGS_Blatt3_Sparse.m