Algorithmenentwurf HA 2

Lukas Brandt: 7011823, Clemens Damke: 7011488, Lukas Giesel: 7011495
21. April 1016

Aufgabe 3

a. Beweis

zu zeigen: (†) Gleichung aus der Aufgabenstellung.

- *Falls:* $k = 1 \implies n = 2$

$$A = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \quad B = \begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{bmatrix}$$

$$C = A \cdot B = \begin{bmatrix} c_{11} = a_{11} \cdot b_{11} + a_{12} \cdot b_{21} & c_{12} = a_{11} \cdot b_{12} + a_{12} \cdot b_{22} \\ c_{21} = a_{21} \cdot b_{11} + a_{22} \cdot b_{21} & c_{22} = a_{21} \cdot b_{12} + a_{22} \cdot b_{22} \end{bmatrix}$$

$\implies$ (†) für $k = 1$, wenn man a_{ij}, b_{ij}, c_{ij} als (1×1)-Matrizen interpretiert.

- *Falls:* $k > 1 \implies n = 2^k$

$$A \cdot B = C = \begin{bmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{bmatrix} = \left[\begin{array}{ccc|ccc} c_{11} & \cdots & c_{1\frac{n}{2}} & c_{1\frac{n}{2}+1} & \cdots & c_{1n} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ c_{\frac{n}{2}1} & \cdots & c_{\frac{n}{2}\frac{n}{2}} & c_{\frac{n}{2}\frac{n}{2}+1} & \cdots & c_{\frac{n}{2}n} \\ \hline c_{\frac{n}{2}+11} & \cdots & c_{\frac{n}{2}+1\frac{n}{2}} & c_{\frac{n}{2}+1\frac{n}{2}+1} & \cdots & c_{\frac{n}{2}+1n} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ c_{n1} & \cdots & c_{n\frac{n}{2}} & c_{n\frac{n}{2}+1} & \cdots & c_{nn} \end{array}\right]$$

$$\begin{aligned} \forall i, j \in \{1, \ldots, n\} : \; c_{ij} &= \sum_{m=1}^{n} a_{im} \cdot b_{mj} = \sum_{m=1}^{\frac{n}{2}} a_{im} \cdot b_{mj} + \sum_{m=\frac{n}{2}+1}^{n} a_{im} \cdot b_{mj} \\ &= \begin{cases} (A_{11} \cdot B_{11})_{ij} + (A_{12} \cdot B_{21})_{ij} & i, j \leq \frac{n}{2} \; (C_{11}) \\ (A_{11} \cdot B_{12})_{i,j-\frac{n}{2}} + (A_{12} \cdot B_{22})_{i,j-\frac{n}{2}} & i \leq \frac{n}{2}, j > \frac{n}{2} \; (C_{12}) \\ (A_{21} \cdot B_{11})_{i-\frac{n}{2},j} + (A_{22} \cdot B_{21})_{i-\frac{n}{2},j} & i > \frac{n}{2}, j \leq \frac{n}{2} \; (C_{21}) \\ (A_{21} \cdot B_{12})_{i-\frac{n}{2},j-\frac{n}{2}} + (A_{22} \cdot B_{22})_{i-\frac{n}{2},j-\frac{n}{2}} & i, j > \frac{n}{2} \; (C_{22}) \end{cases} \\ &= \begin{cases} (A_{11} \cdot B_{11} + A_{12} \cdot B_{21})_{ij} & i, j \leq \frac{n}{2} \; (C_{11}) \\ (A_{11} \cdot B_{12} + A_{12} \cdot B_{22})_{i,j-\frac{n}{2}} & i \leq \frac{n}{2}, j > \frac{n}{2} \; (C_{12}) \\ (A_{21} \cdot B_{11} + A_{22} \cdot B_{21})_{i-\frac{n}{2},j} & i > \frac{n}{2}, j \leq \frac{n}{2} \; (C_{21}) \\ (A_{21} \cdot B_{12} + A_{22} \cdot B_{22})_{i-\frac{n}{2},j-\frac{n}{2}} & i, j > \frac{n}{2} \; (C_{22}) \end{cases} \end{aligned}$$

$\implies$ (†), da die vier Fälle jeweils die Einträge genau einer Submatrix C_{ij} von C beschreiben.

■

Es werden pro Submatrix von C zwei Multiplikationen und eine Addition benötigt. Zur Multiplikation zweier Matrizen müssen vier Submatrizen berechnet und anschließend zusammengefügt werden. Insgesamt werden also 8 Multiplikationen und 4 Additionen benötigt.

b. Algorithmus

Algorithm 1 Matrizenmultiplikation

Require: $\exists k \in \mathbb{N}_0, n := 2^k : A = \begin{bmatrix} a_{11} & \cdots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{n1} & \cdots & a_{nn} \end{bmatrix} \wedge B = \begin{bmatrix} b_{11} & \cdots & b_{1n} \\ \vdots & \ddots & \vdots \\ b_{n1} & \cdots & b_{nn} \end{bmatrix}$

function M-MULTIPLY(A, B)
    if $n = 1$ **then**
        return $a_{11} \cdot b_{11}$ ▷ $\mathcal{O}(1)$

    $A_{11}, A_{12}, A_{21}, A_{22} \leftarrow (\frac{n}{2} \times \frac{n}{2})$-Submatrizen von A ▷ $\mathcal{O}(n^2)$
    $B_{11}, B_{12}, B_{21}, B_{22} \leftarrow (\frac{n}{2} \times \frac{n}{2})$-Submatrizen von B ▷ $\mathcal{O}(n^2)$

    for all $i, j \in \{1, 2\}$ **do**
        $T_1 \leftarrow$ M-MULTIPLY(A_{i1}, B_{1j}) ▷ 8 Aufrufe pro M-MULTIPLY-Call.
        $T_2 \leftarrow$ M-MULTIPLY(A_{i2}, B_{2j}) ▷ 8 Aufrufe pro M-MULTIPLY-Call.
        $C_{ij} \leftarrow$ M-ADD(T_1, T_2) ▷ $\mathcal{O}(n^2)$

    return $\begin{bmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{bmatrix}$ ▷ $\mathcal{O}(n^2)$

function M-ADD(A, B)
    for all $i, j \in \{1, \dots, n\}$ **do**
        $c_{ij} \leftarrow a_{ij} + b_{ij}$

    return $\begin{bmatrix} c_{11} & \cdots & c_{1n} \\ \vdots & \ddots & \vdots \\ c_{n1} & \cdots & c_{nn} \end{bmatrix}$

Laufzeit:

$$T(n) = 8c \cdot T(\frac{n}{2}) + c \cdot n^2$$

$$\underset{\text{Mastertheorem } a>b}{\Longrightarrow} T(n) \leq \frac{8}{8-2} c \cdot n^{\log_2 8} = \frac{4}{3} c \cdot n^3 = \mathcal{O}(n^3)$$