

Übung 10 – Dynamische Programmierung

Beispiel: Minimierung der Anzahl skalarer Multiplikationen bei Matrixketten.

$$(A_1 \times A_2 \times \dots \times A_n)$$

$$(A_1 \times (A_2 \times (A_3 \times A_4)))$$

$$(A_1 \times ((A_2 \times A_3) \times A_4))$$

$$((A_1 \times A_2) \times (A_3 \times A_4))$$

$$((A_1 \times (A_2 \times A_3)) \times A_4)$$

$$(((A_1 \times A_2) \times A_3) \times A_4)$$

02.02.12

Übung 10 - Dynamische Programmierung

1

Algorithmische Idee

- **Splitte** das Problem $A_{1..n}$ in zwei Subprobleme $(A_1 A_{i+1} \dots A_k)$ und $(A_{k+1} A_{k+2} \dots A_n)$. Wähle den Splittpunkt k , der die Anzahl der skalaren Multiplikationen minimiert.
- **Definiere das Optimierungskriterium.** Bewerte die verschiedenen Subproblemlösungen.
- **Berechne die Werte der Subproblemlösungen**
- **Bottom-Up: Kombiniere** kleine Subproblemlösungen zu größeren.

Minimum von $A_{1..2}$ ist $(A_1 \times A_2)$

Minimum von $A_{2..3}$ ist $(A_2 \times A_3)$

Minimum von $A_{1..3}$ ist $((A_1 \times A_2) \times A_3)$ oder $(A_1 \times (A_2 \times A_3))$

02.02.12

Übung 10 - Dynamische Programmierung

2

Minimierungskriterium $m[i, j]$

Sei $m[i, j]$ die minimale **Anzahl skalarer Multiplikationen** für die Berechnung der **Matrix** $A_{i..j}$

Originalproblem: $A_{1..n}$ mit minimalen Kosten $m[1, n]$

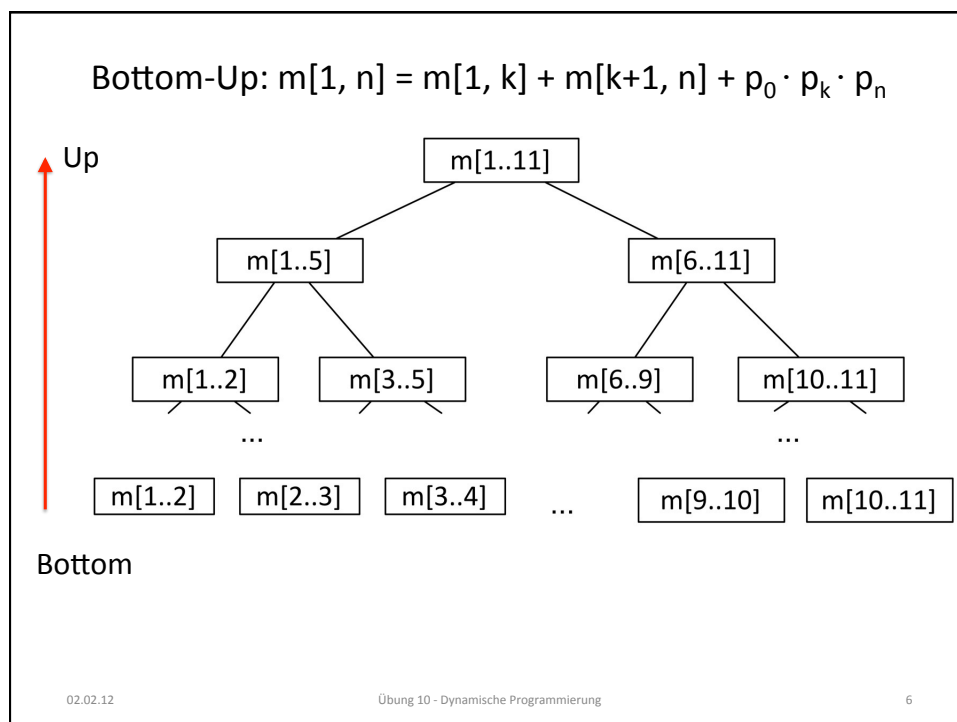
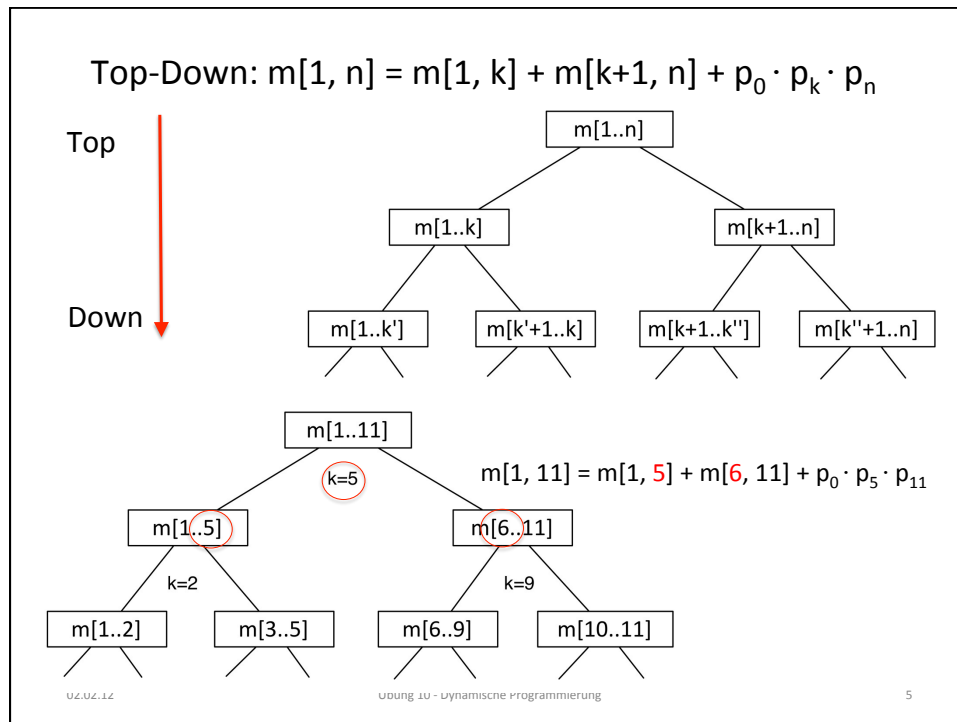
Dimensionen der Matrixkette $A_1 \times A_2 \times \dots \times A_n$: $p_0, p_1, p_2, \dots, p_n$

$$m[i, j] = m[i, k] + m[k+1, j] + p_{i-1} \cdot p_k \cdot p_j$$

Rekursive Lösung (Top-Down)

$$m[1, n] = \begin{cases} 0 & \text{if } n=1 \\ \min_{1 \leq k < n} \{m[1, k] + m[k+1, n] + p_0 \cdot p_k \cdot p_n\} & \text{sonst} \end{cases}$$

$$m[i, j] = \begin{cases} 0 & \text{if } i = j \\ \min_{i \leq k < j} \{m[i, k] + m[k+1, j] + p_{i-1} \cdot p_k \cdot p_j\} & \text{if } i < j \end{cases}$$



Rechnung: $m[1,3]$

$k=1$

$$\begin{aligned} m[1,3] &= \min \{ m[1,1] + m[2,3] + 30 \cdot 35 \cdot 5 \\ &\quad m[1,2] + m[3,3] + 30 \cdot 15 \cdot 5 \} \\ &= \min \{ 2625 + 5250, 15750 + 2625 \} \\ &= 7875 \end{aligned}$$

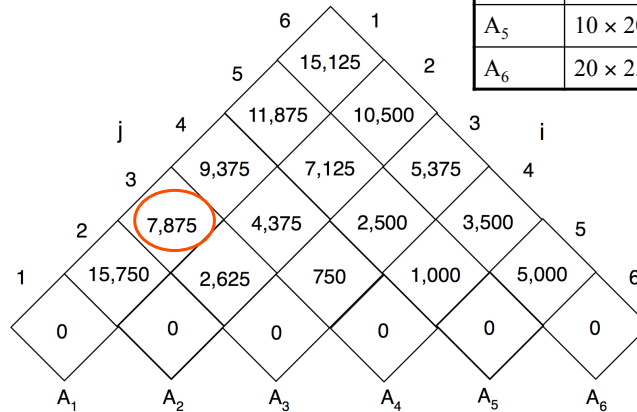
$$(A_1 \times A_2) \times A_3$$

$$30 \times 15 \times 5$$

$$A_1 \times (A_2 \times A_3)$$

$$30 \times 35 \times 5$$

Matrix	Dim.
A_1	30×35
A_2	35×15
A_3	15×5
A_4	5×10
A_5	10×20
A_6	20×25



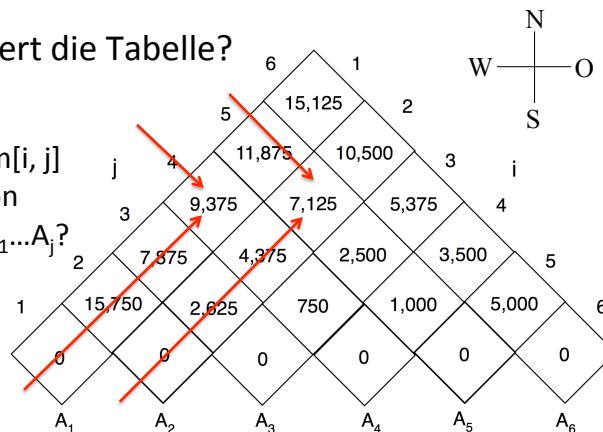
02.02.12

Übung 10 - Dynamische Programmierung

7

Welche Infos liefert die Tabelle?

Wie findet man die minimalen Kosten $m[i, j]$ für die Multiplikation einer Subkette $A_i A_{i+1} \dots A_j$?



Suche die Kreuzung der in Nordost-Richtung verlaufenden Zeilen von A_i und der in Nordwest-Richtung verlaufenden Zeilen von A_j . Beispiele:

$$A_1 \times A_2 \times A_3 \times A_4 = ? \quad A_2 \times A_3 \times A_4 \times A_5 = ? \quad A_1 \dots A_6 = ?$$

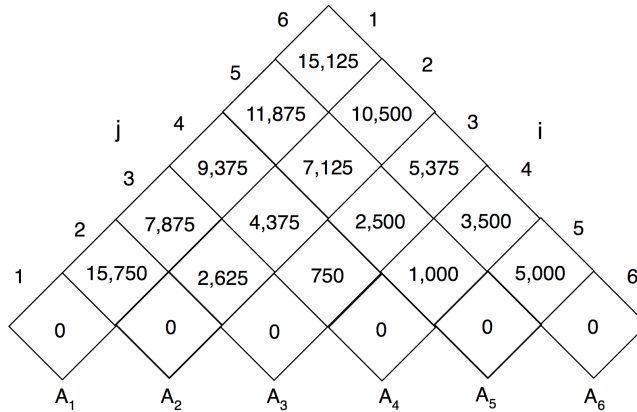
02.02.12

Übung 10 - Dynamische Programmierung

8

Aufgabe: Berechne $m[2,5]$!

Matrix	Dim.
A_1	30×35
A_2	35×15
A_3	15×5
A_4	5×10
A_5	10×20
A_6	20×25



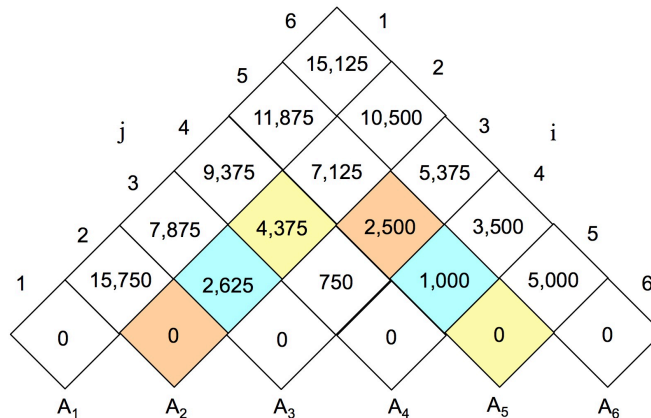
$$m[2,5] = \min_{i \leq k < j} \{m[2,k] + m[k+1,5] + p_1 \cdot p_k \cdot p_5\}$$

02.02.12

Übung 10 - Dynamische Programmierung

9

Lösung:



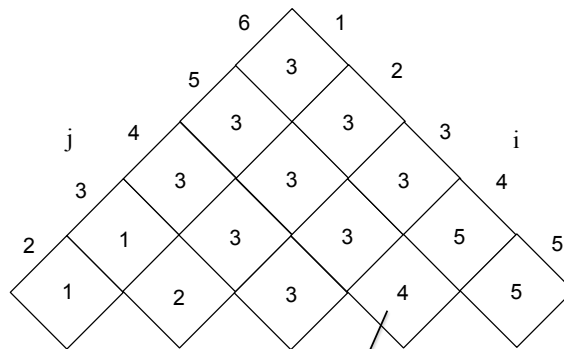
$$m[2,5] = \min \begin{cases} m[2,2] + m[3,5] + p_1 p_2 p_5 = 0 + 2500 + 35 \cdot 15 \cdot 20 = 13000 \\ m[2,3] + m[4,5] + p_1 p_3 p_5 = 2625 + 1000 + 35 \cdot 5 \cdot 20 = 7125 \\ m[2,4] + m[5,5] + p_1 p_4 p_5 = 4375 + 0 + 35 \cdot 10 \cdot 20 = 11375 \end{cases}$$

02.02.12

Übung 10 - Dynamische Programmierung

10

Splittpunkttafel s[k]



$((A_1 (A_2 A_3)) ((A_4 A_5) A_6))$
 $s[1,3] = 1 \quad s[1,6] = 3 \quad s[4,6] = 5$