CS F364: Design & Analysis of Algorithm

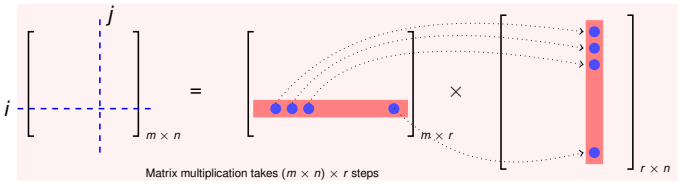
12 Matrix Chain Muliplication

Dr. Kamlesh Tiwari
Assistant Professor, Department of CSIS,
BITS Pilani, Pilani Campus, Rajasthan-333031 INDIA

Feb 12, 2021 ONLINE (Campus @ BITS-Pilani Jan-May 2021)

http://ktiwari.in/algo

Matrix Multiplication



- To multiply three matrices A, B and C of size $u \times v$, $v \times w$ and $w \times z$ respectively, one can do it in two ways
  1. $(A \times B) \times C$: takes $u \times v \times w + u \times w \times z$ steps
  2. $A \times (B \times C)$: takes $u \times v \times z + v \times w \times z$ steps
- For $(u, v, w, z) = (5, 1, 3, 10)$ it is 165 and 80 respectively

How to determine minimum steps for $(v_1, v_2, v_3, ..., v_n)$

Matrix-chain multiplication

- Given a sequence (chain) $< A_1, A_2, ..., A_n >$
- Wish to compute matrix product in minimum number of steps.
- **Parenthesize** $< A_1, A_2, A_3, A_4 >$
  $A_1(A_2(A_3A_4))$, $A_1((A_2A_3)A_4)$, $(A_1A_2)(A_3A_4)$, $(A_1(A_2A_3))A_4$, $((A_1A_2)A_3)A_4$ There are 5 ways
- Let $P(n)$ be the number of ways to parenthesize n matrices then

$$P(n) = \begin{cases} 1 & \text{if } n = 1 \\ \sum_{k=1}^{n-1} P(k) \times P(n-k) & \text{otherwise} \end{cases}$$

- Its solution is Catalan number $P(n) = C(n-1)$, where

$$C(n) = \frac{1}{n+1}\,^{2n}C_n = \Omega(4^n/n^{3/2})$$

Matrix-chain multiplication

- Let matrix A_i has dimension $p_{i-1} \times p_i$ and the sequence $< p_0, p_1, ..., p_n >$ represents all input matrices dimensions
- DP want 1) Optimal substructure, 2) Overlapping subproblems
- DP maintains two matrices m and s as below

Algorithm 1: Matrix-Chain-Multiply(A, s, i, j)

```
1 if j > i then
2     X = Matrix-Chain-Multiply( A, s, i, s[i][j])
3     Y = Matrix-Chain-Multiply( A, s, s[i][j]+1, j)
4     return X×Y
5 else
6     return A_i
```

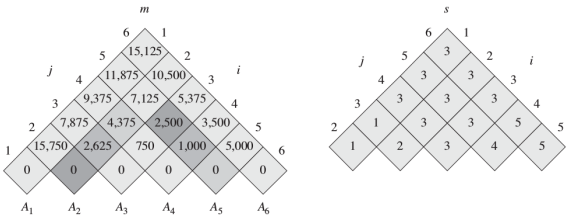

Matrix-chain multiplication

Algorithm 2: Matrix-Chain-Order(p)

```
1  n = length[p] -1
2  for i = 1 to n do
3      m[i][i] = 0
4  for l = 2 to n do
5      for i = 1 to n-l+1 do
6          j = i + l - 1
7          m[i][j] = ∞
8          for k = i to j-1 do
9              q = m[i][k] + m [k+1][j] + p_{i-1} p_k p_j
10             if q < m[i][j] then
11                 m[i][j] = q
12                 s[i][j] = k
```


Matrix-chain multiplication

- Consider $p =< 30, 35, 15, 5, 10, 20, 25 >$



Thank You!

Thank you very much for your attention! (Reference[1])

Queries ?

[1] Book - *Introduction to Algorithm*, By THOMAS H. CORMEN, CHARLES E. LEISERSON, RONALD L. RIVEST, CLIFFORD STEIN