Lecture 14: Chapter 3 Part 2

Dynamic Programming

CS3310

Chained Matrix Multiplication

1	2	3
5	5	6

✖

7	8	9	1
2	3	4	5
6	7	8	9

=

What size matrix results from this multiplication?

How many elementary multiplications take place?

Chained Matrix Multiplication

$$\begin{bmatrix} 1 & 2 & 3 \\ 5 & 5 & 6 \end{bmatrix} \times \begin{bmatrix} 7 & 8 & 9 & 1 \\ 2 & 3 & 4 & 5 \\ 6 & 7 & 8 & 9 \end{bmatrix} = \begin{bmatrix} 29 & 35 & 41 & 38 \\ 74 & 89 & 104 & 83 \end{bmatrix}$$

When we multiply a 2×3 matrix by a 3×4 matrix, the result is a 2×4 matrix.

It takes 3 elementary multiplications to compute each value in the resulting matrix.

Chained Matrix Multiplication

$$\begin{bmatrix} 1 & 2 & 3 \\ 5 & 5 & 6 \end{bmatrix} \times \begin{bmatrix} 7 & 8 & 9 & 1 \\ 2 & 3 & 4 & 5 \\ 6 & 7 & 8 & 9 \end{bmatrix} = \begin{bmatrix} 29 & 35 & 41 & 38 \\ 74 & 89 & 104 & 83 \end{bmatrix}$$

When we multiply a 2×3 matrix by a 3×4 matrix, the result is a 2×4 matrix.

It takes 3 elementary multiplications to compute each value in the resulting matrix.
i.e. $1 \times 7 + 2 \times 2 + 3 \times 6 = 29$

Chained Matrix Multiplication

1	2	3
5	5	6

✖

7	8	9	1
2	3	4	5
6	7	8	9

=

29	35	41	38
74	89	104	83

It takes 3 elementary multiplications to compute each value in the resulting matrix.

- There are 8 values in the final product (2×4).
- Each value requires 3 multiplications to acquire.
- Therefore, the number of elementary multiplications is $2 \times 4 \times 3 = \mathbf{24}$

Multiplying an $i \times j$ matrix by a $j \times k$ matrix requires

- $i \times j \times k$ elementary multiplications.

Chained Matrix Multiplication

Consider the multiplying of the following four matrices of various dimensions:

$$\begin{array}{ccccccc} A & \times & B & \times & C & \times & D \\ 20 \times 2 & & 2 \times 30 & & 30 \times 12 & & 12 \times 8 \end{array}$$

- In what order do we have to multiply these matrices?

Chained Matrix Multiplication

Consider the multiplying of the following four matrices of various dimensions:

$$\begin{array}{ccccccc} A & \times & B & \times & C & \times & D \\ 20 \times 2 & & 2 \times 30 & & 30 \times 12 & & 12 \times 8 \end{array}$$

- **Note**: matrix multiplication is associative.
  - i.e. The order in which we multiply them produces the same result:
    - A(B(CD)) = (AB)(CD) = A((BC)D), etc
- There are *five* ways to multiply four matrices.
- However, the order in which they are multiplied can greatly affect the # of elementary multiplications performed.

Chained Matrix Multiplication

A	B	C	D
20×2	2×30	30×12	12×8

Let's calculate how many elementary multiplications A(B(CD)) takes:

➢ A(B(CD))

- Multiply $C \times D$ first.
  - $30 \times 12 \times 8$ elementary multiplications
  - Results in a 30×8 matrix, E

Chained Matrix Multiplication

A	B	E
20×2	2×30	30×8

Let’s calculate how many elementary multiplications A(B(CD)) takes:

➢ A(BE)

- Multiply C × D first.
  - $30 \times 12 \times 8$ elementary multiplications
  - Results in a 30×8 matrix, E
- Multiply B × E next.
  - $2 \times 30 \times 8$ elementary multiplications
  - Results in a 2×8 matrix, F

Chained Matrix Multiplication

$$\begin{array}{cc} A & F \\ 20 \times 2 & 2 \times 8 \end{array}$$

Let's calculate how many elementary multiplications A(B(CD)) takes:

- ➢ AF
- Multiply $C \times D$ first.
  - $30 \times 12 \times 8$ elementary multiplications
  - Results in a 30×8 matrix, E
- Multiply $B \times E$ next.
  - $2 \times 30 \times 8$ elementary multiplications
  - Results in a 2×8 matrix, F
- Multiply $A \times F$ last.
  - $20 \times 2 \times 8$ elementary multiplications
  - Results in a 20×8 matrix

Chained Matrix Multiplication

We end up with $(30 \times 12 \times 8) + (2 \times 30 \times 8) + (20 \times 2 \times 8)$ or **3,680** elementary multiplications.

Chained Matrix Multiplication

$$\begin{array}{ccccccc} A & \times & B & \times & C & \times & D \\ 20 \times 2 & & 2 \times 30 & & 30 \times 12 & & 12 \times 8 \end{array}$$

Multiplying the arrays in different orders results in a different # of multiplications.

$A(B(CD)) = (30 \times 12 \times 8) + (2 \times 30 \times 8) + (20 \times 2 \times 8) = 3{,}680$

$(AB)(CD) = (20 \times 2 \times 30) + (30 \times 12 \times 8) + (20 \times 30 \times 8) = 8{,}880$

$A((BC)D) = (2 \times 30 \times 12) + (2 \times 12 \times 8) + (20 \times 2 \times 8) = \mathbf{1{,}232}$

$((AB)C)D = (20 \times 2 \times 30) + (20 \times 30 \times 12) + (20 \times 12 \times 8) = 10{,}320$

$(A(BC))D = (2 \times 30 \times 12) + (20 \times 2 \times 12) + (20 \times 12 \times 8) = 3{,}120$

The third order is clearly optimal.

Chained Matrix Multiplication

- Suppose we want to multiply n matrices as efficiently as possible.
  - We want an algorithm that determines the optimal multiplication order.
- This optimal order depends on the dimensions of each matrix.
  - Therefore, we pass the dimensions of each matrix to the algorithm.
- A brute-force algorithm considers every possible order and chooses the minimum, as we previously did.

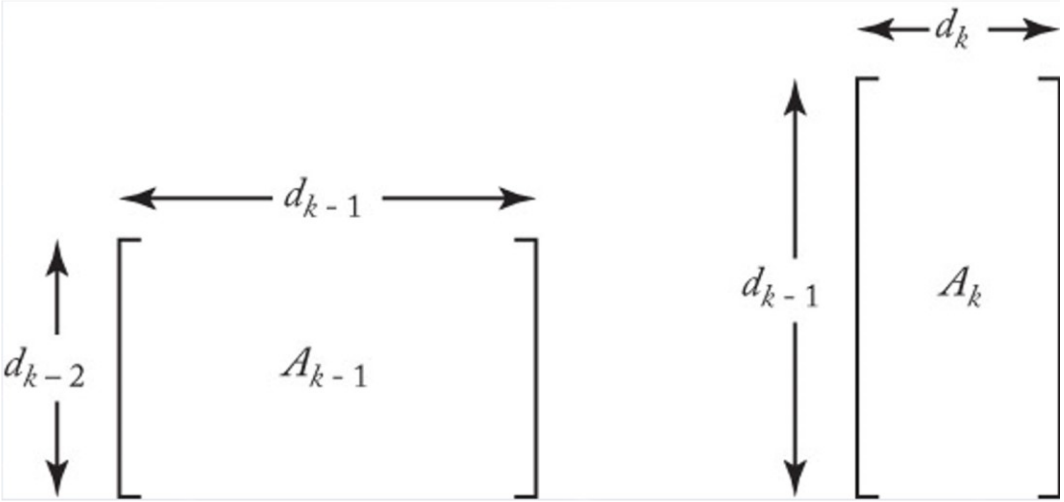
Chained Matrix Multiplication

We can instead use dynamic programming to solve this problem much more efficiently!

- We will use a two dimensional array M to construct our solution:
  - $M[i][j]$ = min # of multiplications to multiply A_i through A_j, if $i < j$
  - $M[i][i] = 0$

Chained Matrix Multiplication

- When we multiply the $(k - 1)_{st}$ matrix, A_{k-1} by the kth matrix, A_k, the number of columns in A_{k-1} *must* equal the number of rows in A_k.
  - Therefore, we can let d_0 be the number of rows in A_1 and d_1 be the number of columns in A_1
  - i.e. d_{k-1} is the number of rows in A_k and d_k is the number of columns in A_k



Chained Matrix Multiplication

Suppose we have the following matrices:

$$
\begin{array}{ccccccccccc}
A_1 & \times & A_2 & \times & A_3 & \times & A_4 & \times & A_5 & \times & A_6 \\
5 \times 2 & & 2 \times 3 & & 3 \times 4 & & 4 \times 6 & & 6 \times 7 & & 7 \times 8 \\
d_0 \quad d_1 & & d_1 \quad d_2 & & d_2 \quad d_3 & & d_3 \quad d_4 & & d_4 \quad d_5 & & d_5 \quad d_6
\end{array}
$$

To multiply A_4, A_5, and A_6, there are two possible orders:

Chained Matrix Multiplication

Suppose we have the following matrices:

$$\begin{array}{ccccccccccc} A_1 & \times & A_2 & \times & A_3 & \times & A_4 & \times & A_5 & \times & A_6 \\ 5 \times 2 & & 2 \times 3 & & 3 \times 4 & & 4 \times 6 & & 6 \times 7 & & 7 \times 8 \\ d_0 \quad d_1 & & d_1 \quad d_2 & & d_2 \quad d_3 & & d_3 \quad d_4 & & d_4 \quad d_5 & & d_5 \quad d_6 \end{array}$$

To multiply A_4, A_5, and A_6, there are two possible orders:

$$(A_4A_5)A_6 = (d_3 \times d_4 \times d_5) + (d_3 \times d_5 \times d_6)$$
$$= 4 \times 6 \times 7 + 4 \times 7 \times 8 = \mathbf{392}$$

$$A_4(A_5A_6) = (d_4 \times d_5 \times d_6) + (d_3 \times d_4 \times d_6)$$
$$= 6 \times 7 \times 8 + 4 \times 6 \times 8 = \mathbf{528}$$

$\therefore M[4][6] = \mathit{min}\ (392, 528) = 392$

Chained Matrix Multiplication

The optimal order for multiplying six matrices must be one of these:

1. $A_1(A_2A_3A_4A_5A_6)$
2. $(A_1A_2)(A_3A_4A_5A_6)$
3. $(A_1A_2A_3)(A_4A_5A_6)$
4. $(A_1A_2A_3A_4)(A_5A_6)$
5. $(A_1A_2A_3A_4A_5)A_6$

The matrices in each set of parentheses are multiplied according to their optimal order

- i.e. in ordering 1, the optimal ordering for $A_2A_3A_4A_5A_6$ is determined using the same algorithm before its result is multiplied by A_1.

Chained Matrix Multiplication

The following equation calculates the cost of multiplying matrix 1 through 2 plus the cost of multiplying matrices 3 through 6 plus the cost of multiplying the two resulting matrices.

- $M[1][2] + M[3][6] + d_0 d_2 d_6$

This equation can be written more generally:

- $M[1][k] + M[k+1][6] + d_0 d_k d_6$

$$\therefore\ M[1][6] = \underset{1 \le k \le 5}{minimum} \left(M[1][k] + M[k+1][6] + d_0 d_k d_6\right)$$

i.e. $min\ (M[1][1] + M[2][6] + d_0 d_1 d_6,$

$M[1][2] + M[3][6] + d_0 d_2 d_6,$

$M[1][3] + M[4][6] + d_0 d_3 d_6,$

$M[1][4] + M[5][6] + d_0 d_4 d_6,$

$M[1][5] + M[6][6] + d_0 d_5 d_6)$

Chained Matrix Multiplication

$$\mathrm{M}[1][6] = \underset{1 \leq k \leq 5}{\mathit{minimum}} \left(\mathrm{M}[1][k] + \mathrm{M}[k+1][6] + d_0 d_k d_6\right)$$

- We also need to find optimal orderings for subsets of the problem (i.e. M[3][5], etc).
  - Therefore, the equation can be written even *more* generally:

$$\mathrm{M}[i][j] = \underset{i \leq k \leq j-1}{\mathit{minimum}} \left(\mathrm{M}[i][k] + \mathrm{M}[k+1][j] + d_{\mathrm{i}-1} d_k d_{\mathrm{j}}\right)$$

- A divide-and-conquer algorithm based on this property is exponential time.
- A dynamic programming algorithm is much more efficient.

Chained Matrix Multiplication

Algorithm Overview:

- Set all entries in the main diagonal to 0.
  - i.e. M[1][1], M[2][2] etc. are 0 since it takes 0 multiplications to multiply A_i through A_i
- Compute the entries in the next diagonal above it.
  - i.e. M[1][2], M[2][3], etc.
- Compute the entries in the next diagonal above the previous one.
  - i.e. M[1][3], M[2][4], etc.
- Continue in this manner until we compute M[1][*j*].
  - This index will contain the minimum # of elementary multiplications required to multiply the matrices 1 through *j* (i.e. all matrices)

Chained Matrix Multiplication Step 1

Compute diagonal 0

Diagonal 0

	1	2	3	4	5	6
1						
2						
3						
4						
5						
6						

$d_0 = 5 \quad d_1 = 2 \quad d_2 = 3 \quad d_3 = 4 \quad d_4 = 6 \quad d_5 = 7 \quad d_6 = 8$

Chained Matrix Multiplication Step 1

Compute diagonal 0

Diagonal 0

$M[i][i] = 0$ for $1 <= i <= 6$

	1	2	3	4	5	6
1	0					
2		0				
3			0			
4				0		
5					0	
6						0

$d_0 = 5 \quad d_1 = 2 \quad d_2 = 3 \quad d_3 = 4 \quad d_4 = 6 \quad d_5 = 7 \quad d_6 = 8$

Chained Matrix Multiplication Step 2

Compute diagonal 1

$M[1][2] =$

Diagonal 1

	1	2	3	4	5	6
1	0					
2		0				
3			0			
4				0		
5					0	
6						0

$d_0 = 5 \quad d_1 = 2 \quad d_2 = 3 \quad d_3 = 4 \quad d_4 = 6 \quad d_5 = 7 \quad d_6 = 8$

Chained Matrix Multiplication Step 2

Compute diagonal 1

$$M[1][2] = \underset{1 \le k \le 1}{minimum} \left(M[1][k] + M[k+1][2] + d_0 d_k d_2\right)$$

$$= M[1][1] + M[2][2] + d_0 d_1 d_2$$
$$= \quad 0 \quad + \quad 0 \quad + \; 5 \times 2 \times 3 = 30$$

- ➢ M[2][3], M[3][4], M[4][5], and M[5][6] are computed in the same way.

$d_0 = 5 \quad d_1 = 2 \quad d_2 = 3 \quad d_3 = 4 \quad d_4 = 6 \quad d_5 = 7 \quad d_6 = 8$

Diagonal 1

	1	2	3	4	5	6
1	0	30				
2		0	24			
3			0	72		
4				0	168	
5					0	336
6						0

Chained Matrix Multiplication Step 3

Compute diagonal 2

$$M[1][3] = \underset{1 \le k \le 2}{minimum} \left(M[1][k] + M[k+1][3] + d_0 d_k d_3\right)$$

$$= min\,(M[1][1] + M[2][3] + d_0 d_1 d_3,$$
$$M[1][2] + M[3][3] + d_0 d_2 d_3)$$
$$= min\,(0 + 24 + 5 \times 2 \times 4,$$
$$30 + 0 + 5 \times 3 \times 4)$$
$$= \mathbf{64}$$

- ➢ M[2][4], M[3][5], and M[4][6] are computed in the same way.

Diagonal 2

	1	2	3	4	5	6
1	0	30	64			
2		0	24	72		
3			0	72	198	
4				0	168	392
5					0	336
6						0

$d_0 = 5 \quad d_1 = 2 \quad d_2 = 3 \quad d_3 = 4 \quad d_4 = 6 \quad d_5 = 7 \quad d_6 = 8$

Chained Matrix Multiplication

Compute diagonal 3

M[1][4] =

Diagonal 3

	1	2	3	4	5	6
1	0	30	64			
2		0	24	72		
3			0	72	198	
4				0	168	392
5					0	336
6						0

$d_0 = 5 \quad d_1 = 2 \quad d_2 = 3 \quad d_3 = 4 \quad d_4 = 6 \quad d_5 = 7 \quad d_6 = 8$

Chained Matrix Multiplication

Compute diagonal 3

$$M[1][4] = \underset{1 \leq k \leq 3}{minimum} (M[1][k] + M[k+1][4] + d_0 d_k d_4)$$

$$= min\,(M[1][1] + M[2][4] + d_0 d_1 d_4,$$
$$M[1][2] + M[3][4] + d_0 d_2 d_4,$$
$$M[1][3] + M[4][4] + d_0 d_3 d_4)$$
$$= min\,(0 + 72 + 5 \times 2 \times 6,$$
$$30 + 72 + 5 \times 3 \times 6,$$
$$64 + 0 + 5 \times 4 \times 6)$$
$$= \mathbf{132}$$

Diagonal 3

	1	2	3	4	5	6
1	0	30	64	132		
2		0	24	72	156	
3			0	72	198	366
4				0	168	392
5					0	336
6						0

$d_0 = 5 \quad d_1 = 2 \quad d_2 = 3 \quad d_3 = 4 \quad d_4 = 6 \quad d_5 = 7 \quad d_6 = 8$

Chained Matrix Multiplication

Compute diagonal 4

M[1][5] =

Diagonal 4

	1	2	3	4	5	6
1	0	30	64	132		
2		0	24	72	156	
3			0	72	198	366
4				0	168	392
5					0	336
6						0

Chained Matrix Multiplication

Compute diagonal 4

$$M[1][5] = \underset{1 \le k \le 4}{minimum} (M[1][k] + M[k+1][5] + d_0 d_k d_5)$$

$$= min\,(M[1][1] + M[2][5] + d_0 d_1 d_5,$$
$$M[1][2] + M[3][5] + d_0 d_2 d_5,$$
$$M[1][3] + M[4][5] + d_0 d_3 d_5,$$
$$M[1][4] + M[5][5] + d_0 d_4 d_5)$$
$$= min\,(0 + 156 + 5 \times 2 \times 7,$$
$$30 + 198 + 5 \times 3 \times 7,$$
$$64 + 168 + 5 \times 4 \times 7,$$
$$132 + 0 + 5 \times 6 \times 7) = \mathbf{226}$$

$d_0 = 5$ $d_1 = 2$ $d_2 = 3$ $d_3 = 4$ $d_4 = 6$ $d_5 = 7$ $d_6 = 8$

Diagonal 4

	1	2	3	4	5	6
1	0	30	64	132	226	
2		0	24	72	156	268
3			0	72	198	366
4				0	168	392
5					0	336
6						0

Chained Matrix Multiplication

Compute diagonal 5

M[1][6] =

Diagonal 5

	1	2	3	4	5	6
1	0	30	64	132	226	
2		0	24	72	156	268
3			0	72	198	366
4				0	168	392
5					0	336
6						0

Chained Matrix Multiplication

Compute diagonal 5

$$M[1][6] = \underset{1 \le k \le 5}{minimum} \left(M[1][k] + M[k+1][6] + d_0 d_k d_6\right)$$

$$\begin{aligned}
&= min\,(M[1][1] + M[2][6] + d_0d_1d_6,\\
&\qquad M[1][2] + M[3][6] + d_0d_2d_6,\\
&\qquad M[1][3] + M[4][6] + d_0d_3d_6,\\
&\qquad M[1][4] + M[5][6] + d_0d_4d_6\\
&\qquad M[1][5] + M[6][6] + d_0d_5d_6)\\
&= min\,(0 + 268 + 5 \times 2 \times 8,\\
&\qquad 30 + 366 + 5 \times 3 \times 8,\\
&\qquad 64 + 392 + 5 \times 4 \times 8,\\
&\qquad 132 + 336 + 5 \times 6 \times 8,\\
&\qquad 226 + 0 + 5 \times 7 \times 8) = \mathbf{348} \text{ (final answer)}
\end{aligned}$$

$d_0 = 5 \quad d_1 = 2 \quad d_2 = 3 \quad d_3 = 4 \quad d_4 = 6 \quad d_5 = 7 \quad d_6 = 8$

Diagonal 5

	1	2	3	4	5	6
1	0	30	64	132	226	348
2		0	24	72	156	268
3			0	72	198	366
4				0	168	392
5					0	336
6						0

Chained Matrix Multiplication

Problem: Determine the minimum # of elementary multiplications needed to multiply n matrices and a multiplication order that produces that number.

Inputs: The # of matrices n, and an array of integers d, indexed from 0 to n, where d[i - 1] × d[i] is the dimension of the ith matrix.

Outputs: `minmult`, the minimum # of elementary multiplications needed to multiply the n matrices; a two-dimensional array P from which the optimal order can be obtained. P has its rows indexed from 1 to n - 1 and its columns indexed from 1 to n. P[i][j] is the point where matrices i through j are split in an optimal order for multiplying i through j.

For example, P[1][5] = 4 would mean that $(A_1A_2A_3A_4)A_5$ is the optimal way to multiply matrices A_1 through A_5

Chained Matrix Multiplication

```
int minmult (int n, const int d[], index P[][])
          int M[1...n][1...n]

          for (index i = 1; i <= n; i++)
                    M[i][i] = 0;               // initialize first diagonal to 0
          for (index diagonal = 1; diagonal <= n - 1; diagonal++)
                    for (index i = 1; i <= n - diagonal; i++)
                    {
                              j = i + diagonal;
```

$$M[i][j] = \min_{i \le k \le j-1} (M[i][k] + M[k+1][j] + d[i-1] * d[k] * d[j]);$$

```
               P[i][j] = a value of k that gave the minimum;
                    }
          return M[1][n];
```

Chained Matrix Multiplication

The optimal order can be obtained from the array P.

- $P[2][5] = 4$ indicates that the optimal order for multiplying A_2 through A_5 has the following factorization: $(A_2A_3A_4)A_5$

	1	2	3	4	5	6
1		1	1	1	1	1
2			2	3	4	5
3				3	4	5
4					4	5
5						5

Chained Matrix Multiplication

We can find the top-level factorization by visiting P[1][n]

- n = 6, P[1][6] = 1
  - $A_1(A_2A_3A_4A_5A_6)$
- Next we determine the factorization of multiplying A_2 - A_6:
  - P[2][6] = 5
  - $(A_2A_3A_4A_5)A_6$
- We now have $A_1((A_2A_3A_4A_5)A_6)$
- To determine the factorization of A_2 - $A_{5:}$
  - P[2][5] = 4
  - $(A_2A_3A_4)A_5$
  - P[2][4] = 3
  - $(A_2A_3)A_4)$
- Final: $A_1((((A_2A_3)A_4)A_5)A_6)$

	1	2	3	4	5	6
1		1	1	1	1	1
2			2	3	4	5
3				3	4	5
4					4	5
5						5

In-Class Exercise

Find the optimal order, and its cost, for evaluating the product of the following matrices:

A_1	$\times$	A_2	$\times$	A_3	$\times$	A_4	$\times$	A_5
10×4		4×5		5×20		20×2		2×50

Show the final arrays M and P.

$$\mathrm{M}[i][j] = \underset{i \le k \le j-1}{\textit{minimum}} \left(\mathrm{M}[i][k] + \mathrm{M}[k+1][j] + d_{i-1} d_k d_j\right)$$

$\mathrm{P}[i][j]$ = the value of k when $\mathrm{M}[i][j]$ is chosen