Matrix Multiplication

I) Conventional method

MATRIX-MULTIPLY (A, B)

```
n = A.rows
Let C be a new n x n matrix
for i = 1 to n
    for j = 1 to n
        c_ij = 0
        for k = 1 to n
            c_ij = c_ij + a_ik · b_kj
return C
```

Time complexity

$\Theta(n^3)$

II) Divide-and-Conquer method

$$A = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}_{2\times 2}, \quad B = \begin{bmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{bmatrix}_{2\times 2}, \quad C = \begin{bmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{bmatrix}_{2\times 2}$$

where,

$$C_{11} = A_{11} B_{11} + A_{12} \cdot B_{21}$$
$$C_{12} = A_{11} B_{12} + A_{12} B_{22}$$
$$C_{21} = A_{21} B_{11} + A_{22} B_{21}$$
$$C_{22} = A_{21} B_{12} + A_{22} B_{22}$$

8 multiplicates of $\frac{n}{2}$-sized numbers and

4 additions (n^2 additions)

$$T(n) = 8T\left(\frac{n}{2}\right) + \Theta(n^2)$$

By master method we get,

$$\boxed{T(n) = \Theta(n^3)}$$ [No improvement].

III) <u>Strassen's method</u> (for $2^n \times 2^n$ matrices only)

$$A = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \qquad B = \begin{bmatrix} e & f \\ g & h \end{bmatrix}$$

We compute seven products as follows :-

$$\begin{aligned}
P_1 &= a(f-h) \\
P_2 &= (a+b)h \\
P_3 &= (c+d)e \\
P_4 &= d(g-e) \\
P_5 &= (a+d)(e+h) \\
P_6 &= (b-d)(g+h) \\
P_7 &= (a-c)(e+f)
\end{aligned}$$

$$C = \begin{bmatrix} P_5+P_6+P_4-P_2 & P_1+P_2 \\ P_3+P_4 & P_1-P_3+P_5-P_7 \end{bmatrix}$$

$$T(n) = 7T\left(\frac{n}{2}\right) + O(n^2)$$

$$\Rightarrow \Theta(n^{\log_2 7}) = \underline{\underline{\Theta(n^{2.8074})}}$$

Scanned with CamScanner