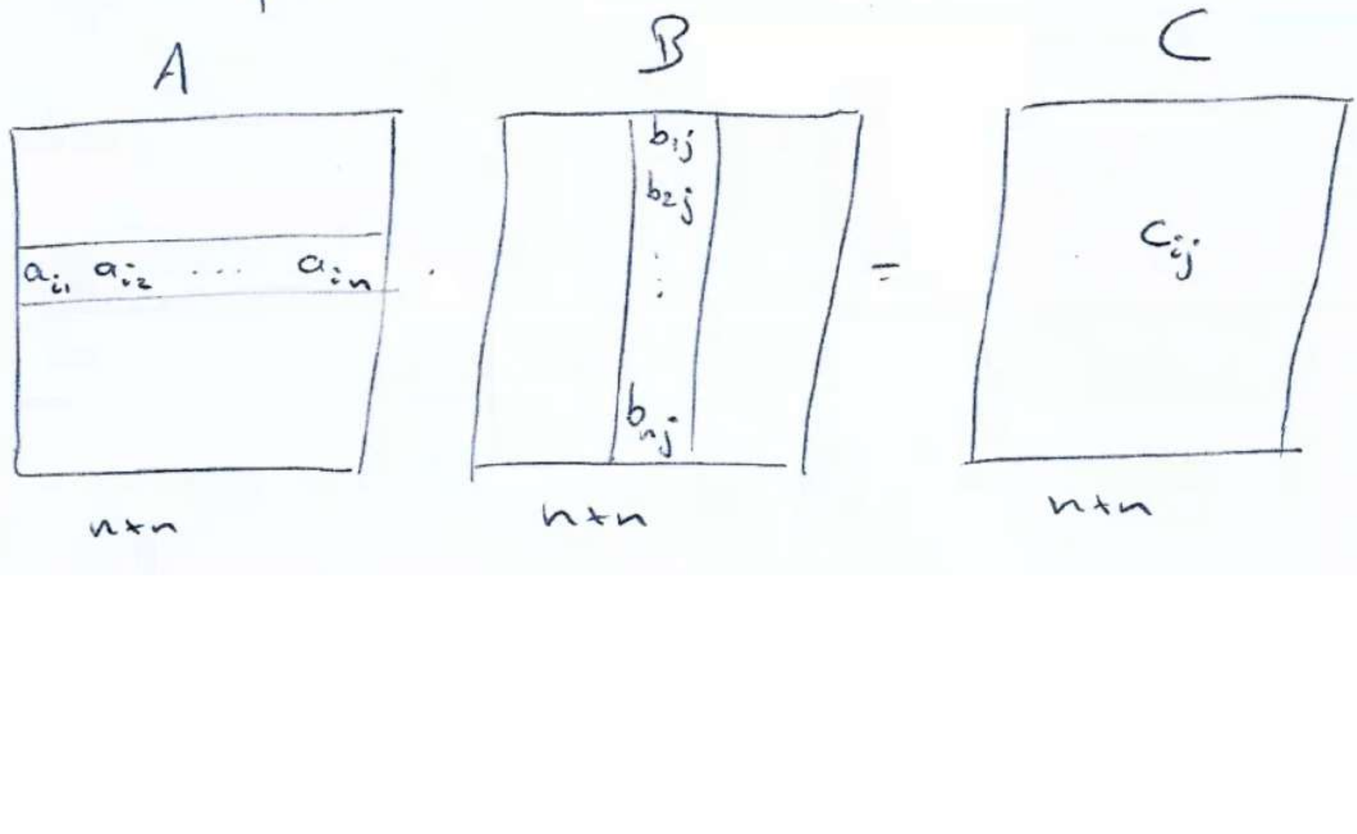
4.2 Matrix Multiplication

Given two $n \times n$ matrices A and B find their product:



Divide-and-Conquer Approach

Given two $n \times n$ matrices A and B, partition the matrices as follows:

$$\overset{A}{\begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}} \cdot \overset{B}{\begin{bmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{bmatrix}} = \overset{C}{\begin{bmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{bmatrix}}$$

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

Matrices of size $\frac{n}{2} \times \frac{n}{2}$

SMMR(A, B)

if $n = 1$ return $a_{11} \cdot b_{11}$

else $C_{11} = \text{SMMR}(A_{11}, B_{11}) + \text{SMMR}(A_{12}, B_{21})$

$C_{12} = \text{SMMR}(A_{11}, B_{12}) + \text{SMMR}(A_{12}, B_{22})$

$C_{21} = \text{SMMR}(A_{21}, B_{11}) + \text{SMMR}(A_{22}, B_{21})$

$C_{22} = \text{SMMR}(A_{21}, B_{21}) + \text{SMMR}(A_{22}, B_{22})$

return C

Let $T(n)$ be running time of SMMR(A, B) on inputs of size $n \times n$.

$T(1) = \Theta(1)$

if $n > 1$, $T(n) = 8T\left(\frac{n}{2}\right) + \Theta(n^2) + \Theta(1)$

$$\therefore \quad T(n) = \begin{cases} \Theta(1) & \text{if } n = 1 \\ 8T\left(\frac{n}{2}\right) + \Theta(n^2) & \text{if } n > 1. \end{cases}$$

We need to solve for $T(n)$:

$$T(n) = \Theta(n^3)$$

Strassen's Algorithm

Given $n \times n$ matrices A and B, partition A, B and C as follows:

$$\overset{A}{\begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}} \cdot \overset{B}{\begin{bmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{bmatrix}} = \overset{C}{\begin{bmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{bmatrix}}$$

Compute 10 $\frac{n}{2} \times \frac{n}{2}$ matrices $S_1, S_2, \ldots, S_{10}$ as on page 80:

$$S_1 = B_{12} - B_{22}$$

$$S_2 = A_{11} + A_{12}$$

$$\vdots$$

$$S_{10} = B_{11} + B_{12}$$

Compute 7 $\frac{n}{2} \times \frac{n}{2}$ matrices $P_1, P_2, \dots, P_7$
as on page 80 :

$$\left.\begin{aligned} P_1 &= A_{11} \cdot S_1 \\ P_2 &= S_2 \cdot B_{22} \\ &\vdots \\ P_7 &= S_9 \cdot S_{10} \end{aligned}\right\} \text{computed recursively}$$

then set

$$C_{11} = P_5 + P_4 - P_2 + P_6$$
$$C_{12} = P_1 + P_2$$
$$C_{21} = P_3 + P_4$$
$$C_{22} = P_5 + P_1 - P_3 - P_7$$

Let $T(n)$ be running time of Strassen's algorithm on inputs of size $n \times n$.

$$T(1) = \Theta(1)$$

if $n > 1$,
$$T(n) = 10\,\Theta\left(\left(\tfrac{n}{2}\right)^2\right) + 7 \cdot T\left(\tfrac{n}{2}\right) + 4\,\Theta\left(\left(\tfrac{n}{2}\right)^2\right) + \Theta(1)$$
$$= 7\,T\left(\tfrac{n}{2}\right) + \Theta(n^2)$$

$$\therefore \quad T(n) = \begin{cases} \Theta(1) & \text{if } n = 1 \\ 7\,T\left(\frac{n}{2}\right) + \Theta(n^2) & \text{if } n > 1 . \end{cases}$$

We need to solve for T(n) :

$$T(n) = \Theta(n^{2,81})$$