Strassen Algorithm

Naive Technique

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

How many operations were performed in this multiplication technique?

4 results with 2 multiplications and 1 addition each. So 8 total multiplications and 4 additions.

In the nxn case, n^2 results, n multiplications and n-1 additions each. So n^3 multiplications and n^2 - n additions.

Strassen Algorithm

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix} \times \begin{bmatrix} e & f \\ g & h \end{bmatrix}$$

$$\begin{bmatrix} p5 + p4 - p2 + p6 & p1 + p2 \\ p3 + p4 & p1 + p5 - p3 - p7 \end{bmatrix}$$

$$\begin{aligned}
p_1 &= (a+d)(e+h) \\
p_2 &= (c+d)(e) \\
p_3 &= (a)(f-h) \\
p_4 &= (d)(-e+g) \\
p_5 &= (a+b)(h) \\
p_6 &= (-a+c)(e+f) \\
p_7 &= (b-d)(g+h)
\end{aligned}$$

We can see that there is now only 7 multiplications but 18 additions.

Strassen Complexity proof by induction

Claim: You can multiply a nxn matrix (where n is 2^k for some k) in $n^{log_2 7}$ multiplications.

Base case was shown on the last slide.

$$\begin{bmatrix} A & B \\ C & D \end{bmatrix} \times \begin{bmatrix} E & F \\ G & H \end{bmatrix}$$

Where entries A-H are matrices of size 2^(k-1).

By our Induction Hypothesis each entry will take 7^(k-1) multiplications. We know a 4x4 matrix takes 7 multiplications so therefore the matrix takes

$$7^k = 7^{log_2 n} = n^{\log_2 7} = n^{2.80735492}$$

Through some optimizations by using tensors as of today the fastest known algorithm runs in. Discovered only in April 2023.

$$n^{2.37188}$$