Fast Matrix Multiplication

Troy Cornelius

University of Puget Sound

April 14th, 2014

Why do we care about fast matrix multiplication?

- Fast matrix multiplication allows us to solve many problems quickly and efficiently.
  - Realtime matrix multiplication is used in computer graphics for
  - Speed is also important in solving larger problems such as matrix decompositions and least squares.

Why do we care about fast matrix multiplication?

- Fast matrix multiplication allows us to solve many problems quickly and efficiently.
  - Realtime matrix multiplication is used in computer graphics for
  - Speed is also important in solving larger problems such as matrix decompositions and least squares.
- In this presentation we will examine a new algorithm for multiplying matrices called **Strassen's Matrix Multiplication**.
- We will compare this to the standard matrix multiplication algorithm by comparing the number of operations.

Standard Matrix Multiplication

- First we will examine the base cases of square matrix multiplication, which involves multiplying two (2×2) matrices.
- The purpose is to review traditional matrix multiplication, and understand how it operates as a computer algorithm

Standard Matrix Multiplication

- First we will examine the base cases of square matrix multiplication, which involves multiplying two (2×2) matrices.
- The purpose is to review traditional matrix multiplication, and understand how it operates as a computer algorithm
- Let A and B be (2×2) matrices, and let $C = M(A, B)$. Under traditional matrix multiplication, we can compute the value of each entry $C_{i,j}$ using **Theorem EMP** [1].

Standard Matrix Multiplication (Continued)

- So we have $\begin{bmatrix} C_{1,1} & C_{1,2} \\ C_{2,1} & C_{2,2} \end{bmatrix} = \begin{bmatrix} A_{1,1} & A_{1,2} \\ A_{2,1} & A_{2,2} \end{bmatrix} \begin{bmatrix} B_{1,1} & B_{1,2} \\ B_{2,1} & B_{2,2} \end{bmatrix}$

Standard Matrix Multiplication (Continued)

- So we have $\begin{bmatrix} C_{1,1} & C_{1,2} \\ C_{2,1} & C_{2,2} \end{bmatrix} = \begin{bmatrix} A_{1,1} & A_{1,2} \\ A_{2,1} & A_{2,2} \end{bmatrix} \begin{bmatrix} B_{1,1} & B_{1,2} \\ B_{2,1} & B_{2,2} \end{bmatrix}$
- Under the standard entrywise definition of matrix multiplication (**Theorem EMP** [3]), to compute the value of C at entry i, j we solve: $C_{i,j} = \sum_{k=1}^{n} A_{i,k} \cdot B_{k,j}$

Standard Matrix Multiplication (Continued)

- So we have $\begin{bmatrix} C_{1,1} & C_{1,2} \\ C_{2,1} & C_{2,2} \end{bmatrix} = \begin{bmatrix} A_{1,1} & A_{1,2} \\ A_{2,1} & A_{2,2} \end{bmatrix} \begin{bmatrix} B_{1,1} & B_{1,2} \\ B_{2,1} & B_{2,2} \end{bmatrix}$
- Under the standard entrywise definition of matrix multiplication (**Theorem EMP** [3]), to compute the value of C at entry i, j we solve: $C_{i,j} = \sum_{k=1}^{n} A_{i,k} \cdot B_{k,j}$
- For the case of $n = 2$, this means that $C_{i,j} = A_{i,1} \cdot B_{1,j} + A_{i,2} \cdot B_{2,j}$, thus

$$\begin{aligned} C_{1,1} &= A_{1,1} \cdot B_{1,1} + A_{1,2} \cdot B_{2,1} \\ C_{1,2} &= A_{1,1} \cdot B_{1,2} + A_{1,2} \cdot B_{2,2} \\ C_{2,1} &= A_{2,1} \cdot B_{1,1} + A_{2,2} \cdot B_{2,1} \\ C_{2,2} &= A_{2,1} \cdot B_{1,2} + A_{2,2} \cdot B_{2,2} \end{aligned}$$

Strassen's Matrix Multiplication

- Now we will define Strassen's Matrix Multiplication for two (2×2) matrices. Define A and B as before, and let $C = S(A, B)$.

$$\begin{bmatrix} C_{1,1} & C_{1,2} \\ C_{2,1} & C_{2,2} \end{bmatrix} = \begin{bmatrix} A_{1,1} & A_{1,2} \\ A_{2,1} & A_{2,2} \end{bmatrix} \begin{bmatrix} B_{1,1} & B_{1,2} \\ B_{2,1} & B_{2,2} \end{bmatrix}$$

Strassen's Matrix Multiplication

- Now we will define Strassen's Matrix Multiplication for two (2×2) matrices. Define A and B as before, and let $C = S(A, B)$.

$$\begin{bmatrix} C_{1,1} & C_{1,2} \\ C_{2,1} & C_{2,2} \end{bmatrix} = \begin{bmatrix} A_{1,1} & A_{1,2} \\ A_{2,1} & A_{2,2} \end{bmatrix} \begin{bmatrix} B_{1,1} & B_{1,2} \\ B_{2,1} & B_{2,2} \end{bmatrix}$$

- Now define the following 7 **scalar** products:

Strassen's Matrix Multiplication

- Now we will define Strassen's Matrix Multiplication for two (2×2) matrices. Define A and B as before, and let $C = S(A, B)$.

$$\begin{bmatrix} C_{1,1} & C_{1,2} \\ C_{2,1} & C_{2,2} \end{bmatrix} = \begin{bmatrix} A_{1,1} & A_{1,2} \\ A_{2,1} & A_{2,2} \end{bmatrix} \begin{bmatrix} B_{1,1} & B_{1,2} \\ B_{2,1} & B_{2,2} \end{bmatrix}$$

- Now define the following 7 **scalar** products:

$$P_1 = (A_{1,1} + A_{2,2}) \cdot (B_{1,1} + B_{2,2})$$

Strassen's Matrix Multiplication

- Now we will define Strassen's Matrix Multiplication for two (2×2) matrices. Define A and B as before, and let $C = S(A, B)$.

$$\begin{bmatrix} C_{1,1} & C_{1,2} \\ C_{2,1} & C_{2,2} \end{bmatrix} = \begin{bmatrix} A_{1,1} & A_{1,2} \\ A_{2,1} & A_{2,2} \end{bmatrix} \begin{bmatrix} B_{1,1} & B_{1,2} \\ B_{2,1} & B_{2,2} \end{bmatrix}$$

- Now define the following 7 **scalar** products:

$$P_1 = (A_{1,1} + A_{2,2}) \cdot (B_{1,1} + B_{2,2})$$
$$P_2 = (A_{2,1} + A_{2,2}) \cdot (B_{1,1})$$

Strassen's Matrix Multiplication

- Now we will define Strassen's Matrix Multiplication for two (2×2) matrices. Define A and B as before, and let $C = S(A, B)$.

$$\begin{bmatrix} C_{1,1} & C_{1,2} \\ C_{2,1} & C_{2,2} \end{bmatrix} = \begin{bmatrix} A_{1,1} & A_{1,2} \\ A_{2,1} & A_{2,2} \end{bmatrix} \begin{bmatrix} B_{1,1} & B_{1,2} \\ B_{2,1} & B_{2,2} \end{bmatrix}$$

- Now define the following 7 **scalar** products:

$$P_1 = (A_{1,1} + A_{2,2}) \cdot (B_{1,1} + B_{2,2})$$
$$P_2 = (A_{2,1} + A_{2,2}) \cdot (B_{1,1})$$
$$P_3 = (A_{1,1}) \cdot (B_{1,2} - B_{2,2})$$

Strassen's Matrix Multiplication

- Now we will define Strassen's Matrix Multiplication for two (2×2) matrices. Define A and B as before, and let $C = S(A, B)$.

$$\begin{bmatrix} C_{1,1} & C_{1,2} \\ C_{2,1} & C_{2,2} \end{bmatrix} = \begin{bmatrix} A_{1,1} & A_{1,2} \\ A_{2,1} & A_{2,2} \end{bmatrix} \begin{bmatrix} B_{1,1} & B_{1,2} \\ B_{2,1} & B_{2,2} \end{bmatrix}$$

- Now define the following 7 **scalar** products:

$$P_1 = (A_{1,1} + A_{2,2}) \cdot (B_{1,1} + B_{2,2})$$
$$P_2 = (A_{2,1} + A_{2,2}) \cdot (B_{1,1})$$
$$P_3 = (A_{1,1}) \cdot (B_{1,2} - B_{2,2})$$
$$P_4 = (A_{2,2}) \cdot (B_{2,1} - B_{1,1})$$

Strassen's Matrix Multiplication

- Now we will define Strassen's Matrix Multiplication for two (2×2) matrices. Define A and B as before, and let $C = S(A, B)$.

$$\begin{bmatrix} C_{1,1} & C_{1,2} \\ C_{2,1} & C_{2,2} \end{bmatrix} = \begin{bmatrix} A_{1,1} & A_{1,2} \\ A_{2,1} & A_{2,2} \end{bmatrix} \begin{bmatrix} B_{1,1} & B_{1,2} \\ B_{2,1} & B_{2,2} \end{bmatrix}$$

- Now define the following 7 **scalar** products:

$$P_1 = (A_{1,1} + A_{2,2}) \cdot (B_{1,1} + B_{2,2})$$
$$P_2 = (A_{2,1} + A_{2,2}) \cdot (B_{1,1})$$
$$P_3 = (A_{1,1}) \cdot (B_{1,2} - B_{2,2})$$
$$P_4 = (A_{2,2}) \cdot (B_{2,1} - B_{1,1})$$
$$P_5 = (A_{1,1} + A_{1,2}) \cdot (B_{2,2})$$

Strassen's Matrix Multiplication

- Now we will define Strassen's Matrix Multiplication for two (2×2) matrices. Define A and B as before, and let $C = S(A, B)$.

$$\begin{bmatrix} C_{1,1} & C_{1,2} \\ C_{2,1} & C_{2,2} \end{bmatrix} = \begin{bmatrix} A_{1,1} & A_{1,2} \\ A_{2,1} & A_{2,2} \end{bmatrix} \begin{bmatrix} B_{1,1} & B_{1,2} \\ B_{2,1} & B_{2,2} \end{bmatrix}$$

- Now define the following 7 **scalar** products:

$$\begin{aligned}
P_1 &= (A_{1,1} + A_{2,2}) \cdot (B_{1,1} + B_{2,2}) \\
P_2 &= (A_{2,1} + A_{2,2}) \cdot (B_{1,1}) \\
P_3 &= (A_{1,1}) \cdot (B_{1,2} - B_{2,2}) \\
P_4 &= (A_{2,2}) \cdot (B_{2,1} - B_{1,1}) \\
P_5 &= (A_{1,1} + A_{1,2}) \cdot (B_{2,2}) \\
P_6 &= (A_{2,1} + A_{1,1}) \cdot (B_{1,1} + B_{1,2})
\end{aligned}$$

Strassen's Matrix Multiplication

- Now we will define Strassen's Matrix Multiplication for two (2×2) matrices. Define A and B as before, and let $C = S(A, B)$.

$$\begin{bmatrix} C_{1,1} & C_{1,2} \\ C_{2,1} & C_{2,2} \end{bmatrix} = \begin{bmatrix} A_{1,1} & A_{1,2} \\ A_{2,1} & A_{2,2} \end{bmatrix} \begin{bmatrix} B_{1,1} & B_{1,2} \\ B_{2,1} & B_{2,2} \end{bmatrix}$$

- Now define the following 7 **scalar** products:

$$\begin{aligned}
P_1 &= (A_{1,1} + A_{2,2}) \cdot (B_{1,1} + B_{2,2}) \\
P_2 &= (A_{2,1} + A_{2,2}) \cdot (B_{1,1}) \\
P_3 &= (A_{1,1}) \cdot (B_{1,2} - B_{2,2}) \\
P_4 &= (A_{2,2}) \cdot (B_{2,1} - B_{1,1}) \\
P_5 &= (A_{1,1} + A_{1,2}) \cdot (B_{2,2}) \\
P_6 &= (A_{2,1} + A_{1,1}) \cdot (B_{1,1} + B_{1,2}) \\
P_7 &= (A_{1,2} - A_{2,2}) \cdot (B_{2,1} + B_{2,2})
\end{aligned}$$

Strassen's Matrix Multiplication (continued)

Observe that:

$$P_1 + P_4 - P_5 + P_7 = (A_{1,1}B_{1,1} + A_{1,1}B_{2,2} + A_{2,2}B_{1,1} + A_{2,2}B_{2,2})$$
$$+ (A_{2,2}B_{2,1} - A_{2,2}B_{1,1}) + (-A_{1,2}B_{2,2} - A_{1,2}B_{2,2})$$
$$+ (A_{1,2}B_{2,1} + A_{1,2}B_{2,2} - A_{2,2}B_{2,1} - A_{2,2}B_{2,2})$$

Strassen's Matrix Multiplication (continued)

Observe that:

$$P_1 + P_4 - P_5 + P_7 = (A_{1,1}B_{1,1} + A_{1,1}B_{2,2} + A_{2,2}B_{1,1} + A_{2,2}B_{2,2})$$
$$+ (A_{2,2}B_{2,1} - A_{2,2}B_{1,1}) + (-A_{1,2}B_{2,2} - A_{1,2}B_{2,2})$$
$$+ (A_{1,2}B_{2,1} + A_{1,2}B_{2,2} - A_{2,2}B_{2,1} - A_{2,2}B_{2,2})$$
$$= A_{1,1}B_{1,1} + A_{1,2}B_{2,1}$$
$$= C_{1,1}$$

Strassen's Matrix Multiplication (continued)

Observe that:

$$
\begin{aligned}
P_1 + P_4 - P_5 + P_7 &= (A_{1,1}B_{1,1} + A_{1,1}B_{2,2} + A_{2,2}B_{1,1} + A_{2,2}B_{2,2}) \\
&+ (A_{2,2}B_{2,1} - A_{2,2}B_{1,1}) + (-A_{1,2}B_{2,2} - A_{1,2}B_{2,2}) \\
&+ (A_{1,2}B_{2,1} + A_{1,2}B_{2,2} - A_{2,2}B_{2,1} - A_{2,2}B_{2,2}) \\
&= A_{1,1}B_{1,1} + A_{1,2}B_{2,1} \\
&= C_{1,1}
\end{aligned}
$$

- Using algebra we can also show that

$$
\begin{aligned}
C_{1,1} &= P_1 + P_4 - P_5 + P_7 \\
C_{1,2} &= P_3 + P_5 \\
C_{2,1} &= P_2 + P_4 \\
C_{2,2} &= P_1 + P_3 - P_2 + P_6
\end{aligned}
$$

Strassen's Multiplication Generalized

- Now we will define Strassen's algorithm for a general case.
- Let A and B be $(m \times m)$ matrices, where $m = 2^q$ for some $q \in \mathbb{N}$. Let $C = S(A, B)$.

Strassen's Multiplication Generalized

- Now we will define Strassen's algorithm for a general case.
- Let A and B be $(m \times m)$ matrices, where $m = 2^q$ for some $q \in \mathbb{N}$. Let $C = S(A, B)$.
- To perform Strassen Multiplication on these matrices, partition both A and B into four equally sized submatrices that correspond to the four corners of the matrix.

Strassen's Multiplication Generalized

- Now we will define Strassen's algorithm for a general case.
- Let A and B be $(m \times m)$ matrices, where $m = 2^q$ for some $q \in \mathbb{N}$. Let $C = S(A, B)$.
- To perform Strassen Multiplication on these matrices, partition both A and B into four equally sized submatrices that correspond to the four corners of the matrix.
- Let $\hat{A}_{1,1}$, $\hat{A}_{1,2}$, $\hat{A}_{2,1}$ and $\hat{A}_{2,2}$ be the four matrices that correspond to these corners, and let B and C have submatrices defined similarly.
- Each of these submatrices have size $(2^{q-1} \times 2^{q-1})$.

Strassen's Multiplication Generalized

- Now we will define Strassen's algorithm for a general case.
- Let A and B be $(m \times m)$ matrices, where $m = 2^q$ for some $q \in \mathbb{N}$. Let $C = S(A, B)$.
- To perform Strassen Multiplication on these matrices, partition both A and B into four equally sized submatrices that correspond to the four corners of the matrix.
- Let $\hat{A}_{1,1}$, $\hat{A}_{1,2}$, $\hat{A}_{2,1}$ and $\hat{A}_{2,2}$ be the four matrices that correspond to these corners, and let B and C have submatrices defined similarly.
- Each of these submatrices have size $(2^{q-1} \times 2^{q-1})$.
- $A = \left(\begin{array}{c|c} \hat{A}_{1,1} & \hat{A}_{1,2} \\ \hline \hat{A}_{2,1} & \hat{A}_{2,2} \end{array}\right)$, $B = \left(\begin{array}{c|c} \hat{B}_{1,1} & \hat{B}_{1,2} \\ \hline \hat{B}_{2,1} & \hat{B}_{2,2} \end{array}\right)$, $C = \left(\begin{array}{c|c} \hat{C}_{1,1} & \hat{C}_{1,2} \\ \hline \hat{C}_{2,1} & \hat{C}_{2,2} \end{array}\right)$

Strassen's Multiplication Generalized (continued)

- Now we can use Strassen's algorithm to multiply the two matrices:

$$C = S(A,B) = S\left(\left(\begin{array}{c|c} \hat{A}_{1,1} & \hat{A}_{1,2} \\ \hline \hat{A}_{2,1} & \hat{A}_{2,2} \end{array}\right), \left(\begin{array}{c|c} \hat{B}_{1,1} & \hat{B}_{1,2} \\ \hline \hat{B}_{2,1} & \hat{B}_{2,2} \end{array}\right)\right)$$

Strassen's Multiplication Generalized (continued)

- Now we can use Strassen's algorithm to multiply the two matrices:

$$C = S(A,B) = S\left(\left(\begin{array}{c|c} \hat{A}_{1,1} & \hat{A}_{1,2} \\ \hline \hat{A}_{2,1} & \hat{A}_{2,2} \end{array}\right), \left(\begin{array}{c|c} \hat{B}_{1,1} & \hat{B}_{1,2} \\ \hline \hat{B}_{2,1} & \hat{B}_{2,2} \end{array}\right)\right)$$

- Now define the following **7 Strassen** products:

Strassen's Multiplication Generalized (continued)

- Now we can use Strassen's algorithm to multiply the two matrices:

$$C = S(A,B) = S\left(\left(\begin{array}{c|c} \hat{A}_{1,1} & \hat{A}_{1,2} \\ \hline \hat{A}_{2,1} & \hat{A}_{2,2} \end{array}\right), \left(\begin{array}{c|c} \hat{B}_{1,1} & \hat{B}_{1,2} \\ \hline \hat{B}_{2,1} & \hat{B}_{2,2} \end{array}\right)\right)$$

- Now define the following **7 Strassen** products:

$$\hat{P}_1 = S\left((\hat{A}_{1,1} + \hat{A}_{2,2}), (\hat{B}_{1,1} + \hat{B}_{2,2})\right)$$

Strassen's Multiplication Generalized (continued)

- Now we can use Strassen's algorithm to multiply the two matrices:

$$C = S(A,B) = S\left(\left(\begin{array}{c|c} \hat{A}_{1,1} & \hat{A}_{1,2} \\ \hline \hat{A}_{2,1} & \hat{A}_{2,2} \end{array}\right), \left(\begin{array}{c|c} \hat{B}_{1,1} & \hat{B}_{1,2} \\ \hline \hat{B}_{2,1} & \hat{B}_{2,2} \end{array}\right)\right)$$

- Now define the following 7 **Strassen** products:

$$\hat{P}_1 = S\Big((\hat{A}_{1,1} + \hat{A}_{2,2}), (\hat{B}_{1,1} + \hat{B}_{2,2})\Big)$$

$$\hat{P}_2 = S\Big((\hat{A}_{2,1} + \hat{A}_{2,2}), (\hat{B}_{1,1})\Big)$$

Strassen's Multiplication Generalized (continued)

- Now we can use Strassen's algorithm to multiply the two matrices:

$$C = S(A,B) = S\left(\left(\begin{array}{c|c} \hat{A}_{1,1} & \hat{A}_{1,2} \\ \hline \hat{A}_{2,1} & \hat{A}_{2,2} \end{array}\right), \left(\begin{array}{c|c} \hat{B}_{1,1} & \hat{B}_{1,2} \\ \hline \hat{B}_{2,1} & \hat{B}_{2,2} \end{array}\right)\right)$$

- Now define the following 7 **Strassen** products:

$$\hat{P}_1 = S\Big((\hat{A}_{1,1} + \hat{A}_{2,2}), (\hat{B}_{1,1} + \hat{B}_{2,2})\Big)$$

$$\hat{P}_2 = S\Big((\hat{A}_{2,1} + \hat{A}_{2,2}), (\hat{B}_{1,1})\Big)$$

$$\hat{P}_3 = S\Big((\hat{A}_{1,1}), (\hat{B}_{1,2} - \hat{B}_{2,2})\Big)$$

Strassen's Multiplication Generalized (continued)

- Now we can use Strassen's algorithm to multiply the two matrices:

$$C = S(A,B) = S\left(\left(\begin{array}{c|c} \hat{A}_{1,1} & \hat{A}_{1,2} \\ \hline \hat{A}_{2,1} & \hat{A}_{2,2} \end{array}\right), \left(\begin{array}{c|c} \hat{B}_{1,1} & \hat{B}_{1,2} \\ \hline \hat{B}_{2,1} & \hat{B}_{2,2} \end{array}\right)\right)$$

- Now define the following 7 **Strassen** products:

$$\hat{P}_1 = S\left((\hat{A}_{1,1} + \hat{A}_{2,2}), (\hat{B}_{1,1} + \hat{B}_{2,2})\right)$$

$$\hat{P}_2 = S\left((\hat{A}_{2,1} + \hat{A}_{2,2}), (\hat{B}_{1,1})\right)$$

$$\hat{P}_3 = S\left((\hat{A}_{1,1}), (\hat{B}_{1,2} - \hat{B}_{2,2})\right)$$

$$\hat{P}_4 = S\left((\hat{A}_{2,2}), (\hat{B}_{2,1} - \hat{B}_{1,1})\right)$$

Strassen's Multiplication Generalized (continued)

- Now we can use Strassen's algorithm to multiply the two matrices:

$$C = S(A,B) = S\left(\left(\begin{array}{c|c} \hat{A}_{1,1} & \hat{A}_{1,2} \\ \hline \hat{A}_{2,1} & \hat{A}_{2,2} \end{array}\right), \left(\begin{array}{c|c} \hat{B}_{1,1} & \hat{B}_{1,2} \\ \hline \hat{B}_{2,1} & \hat{B}_{2,2} \end{array}\right)\right)$$

- Now define the following 7 **Strassen** products:

$$\hat{P}_1 = S\left((\hat{A}_{1,1} + \hat{A}_{2,2}), (\hat{B}_{1,1} + \hat{B}_{2,2})\right)$$

$$\hat{P}_2 = S\left((\hat{A}_{2,1} + \hat{A}_{2,2}), (\hat{B}_{1,1})\right)$$

$$\hat{P}_3 = S\left((\hat{A}_{1,1}), (\hat{B}_{1,2} - \hat{B}_{2,2})\right)$$

$$\hat{P}_4 = S\left((\hat{A}_{2,2}), (\hat{B}_{2,1} - \hat{B}_{1,1})\right)$$

$$\hat{P}_5 = S\left((\hat{A}_{1,1} + \hat{A}_{1,2}), (\hat{B}_{2,2})\right)$$

Strassen's Multiplication Generalized (continued)

- Now we can use Strassen's algorithm to multiply the two matrices:

$$C = S(A,B) = S\left(\left(\begin{array}{c|c} \hat{A}_{1,1} & \hat{A}_{1,2} \\ \hline \hat{A}_{2,1} & \hat{A}_{2,2} \end{array} \right), \left(\begin{array}{c|c} \hat{B}_{1,1} & \hat{B}_{1,2} \\ \hline \hat{B}_{2,1} & \hat{B}_{2,2} \end{array} \right) \right)$$

- Now define the following 7 **Strassen** products:

$$\hat{P}_1 = S\Big((\hat{A}_{1,1} + \hat{A}_{2,2}), (\hat{B}_{1,1} + \hat{B}_{2,2}) \Big)$$

$$\hat{P}_2 = S\Big((\hat{A}_{2,1} + \hat{A}_{2,2}), (\hat{B}_{1,1}) \Big)$$

$$\hat{P}_3 = S\Big((\hat{A}_{1,1}), (\hat{B}_{1,2} - \hat{B}_{2,2}) \Big)$$

$$\hat{P}_4 = S\Big((\hat{A}_{2,2}), (\hat{B}_{2,1} - \hat{B}_{1,1}) \Big)$$

$$\hat{P}_5 = S\Big((\hat{A}_{1,1} + \hat{A}_{1,2}), (\hat{B}_{2,2}) \Big)$$

$$\hat{P}_6 = S\Big((\hat{A}_{2,1} + \hat{A}_{1,1}), (\hat{B}_{1,1} + \hat{B}_{1,2}) \Big)$$

Strassen's Multiplication Generalized (continued)

- Now we can use Strassen's algorithm to multiply the two matrices:

$$C = S(A,B) = S\left(\left(\begin{array}{c|c}\hat{A}_{1,1} & \hat{A}_{1,2} \\ \hline \hat{A}_{2,1} & \hat{A}_{2,2}\end{array}\right), \left(\begin{array}{c|c}\hat{B}_{1,1} & \hat{B}_{1,2} \\ \hline \hat{B}_{2,1} & \hat{B}_{2,2}\end{array}\right)\right)$$

- Now define the following 7 **Strassen** products:

$$\hat{P}_1 = S\Big((\hat{A}_{1,1} + \hat{A}_{2,2}), (\hat{B}_{1,1} + \hat{B}_{2,2})\Big)$$

$$\hat{P}_2 = S\Big((\hat{A}_{2,1} + \hat{A}_{2,2}), (\hat{B}_{1,1})\Big)$$

$$\hat{P}_3 = S\Big((\hat{A}_{1,1}), (\hat{B}_{1,2} - \hat{B}_{2,2})\Big)$$

$$\hat{P}_4 = S\Big((\hat{A}_{2,2}), (\hat{B}_{2,1} - \hat{B}_{1,1})\Big)$$

$$\hat{P}_5 = S\Big((\hat{A}_{1,1} + \hat{A}_{1,2}), (\hat{B}_{2,2})\Big)$$

$$\hat{P}_6 = S\Big((\hat{A}_{2,1} + \hat{A}_{1,1}), (\hat{B}_{1,1} + \hat{B}_{1,2})\Big)$$

$$\hat{P}_7 = S\Big((\hat{A}_{1,2} - \hat{A}_{2,2}), (\hat{B}_{2,1} + \hat{B}_{2,2})\Big)$$

Strassen's Multiplication Generalized (continued)

- Recall that earlier we proved that $C_{1,1} = P_1 + P_4 - P_5 + P_7$.
- In the base case of Strassen's algorithm, $C_{1,1}$ was a scalar that was the top-left corner of C, and each P_i was a product of scalar entries.

Strassen's Multiplication Generalized (continued)

- Recall that earlier we proved that $C_{1,1} = P_1 + P_4 - P_5 + P_7$.
- In the base case of Strassen's algorithm, $C_{1,1}$ was a scalar that was the top-left corner of C, and each P_i was a product of scalar entries.
- In the generalized case we defined $C = S(A, B)$:

$$C = \left(\begin{array}{c|c} \hat{C}_{1,1} & \hat{C}_{1,2} \\ \hline \hat{C}_{2,1} & \hat{C}_{2,2} \end{array}\right) = S\left(\left(\begin{array}{c|c} \hat{A}_{1,1} & \hat{A}_{1,2} \\ \hline \hat{A}_{2,1} & \hat{A}_{2,2} \end{array}\right), \left(\begin{array}{c|c} \hat{B}_{1,1} & \hat{B}_{1,2} \\ \hline \hat{B}_{2,1} & \hat{B}_{2,2} \end{array}\right)\right)$$

- Similar to the base case, we can prove that:

$$\begin{aligned} \hat{C}_{1,1} &= \hat{P}_1 + \hat{P}_4 - \hat{P}_5 + \hat{P}_7 \\ \hat{C}_{1,2} &= \hat{P}_3 + \hat{P}_5 \\ \hat{C}_{2,1} &= \hat{P}_2 + \hat{P}_4 \\ \hat{C}_{2,2} &= \hat{P}_1 + \hat{P}_3 - \hat{P}_2 + \hat{P}_6 \end{aligned}$$

Strassen's Multiplication Summary

- In Strassen's multiplication algorithm, we took matrices of size $(2^q \times 2^q)$ and divided them into four submatrices of size $(2^{q-1} \times 2^{q-1})$.

$$\left(\begin{array}{c|c} \hat{C}_{1,1} & \hat{C}_{1,2} \\ \hline \hat{C}_{2,1} & \hat{C}_{2,2} \end{array}\right) = S\left(\left(\begin{array}{c|c} \hat{A}_{1,1} & \hat{A}_{1,2} \\ \hline \hat{A}_{2,1} & \hat{A}_{2,2} \end{array}\right), \left(\begin{array}{c|c} \hat{B}_{1,1} & \hat{B}_{1,2} \\ \hline \hat{B}_{2,1} & \hat{B}_{2,2} \end{array}\right)\right)$$

Strassen's Multiplication Summary

- In Strassen's multiplication algorithm, we took matrices of size $(2^q \times 2^q)$ and divided them into four submatrices of size $(2^{q-1} \times 2^{q-1})$.

$$\left(\begin{array}{c|c} \hat{C}_{1,1} & \hat{C}_{1,2} \\ \hline \hat{C}_{2,1} & \hat{C}_{2,2} \end{array}\right) = S\left(\left(\begin{array}{c|c} \hat{A}_{1,1} & \hat{A}_{1,2} \\ \hline \hat{A}_{2,1} & \hat{A}_{2,2} \end{array}\right), \left(\begin{array}{c|c} \hat{B}_{1,1} & \hat{B}_{1,2} \\ \hline \hat{B}_{2,1} & \hat{B}_{2,2} \end{array}\right)\right)$$

- Our goal was to express the matrices $\hat{C}_{1,1}, \hat{C}_{2,1}, \hat{C}_{1,2}$ and $\hat{C}_{2,2}$ in terms of the submatrices for A and B. However, these terms were linear combinations of even more Strassen products.
- Strassen's product is a recursive algorithm. At each "level" of the algorithm, we are taking 7 more Strassen products, which is one for each $\hat{P}_i$.
- In this general example, matrices are $(2^q \times 2^q)$, which means that the algorithm will have $\log_2 2^q = q$ levels until its completion. On the final level, the algorithm will proceed as the base case, giving us a scalar answer rather than a recursive answer.

Strassen's Multiplication Example 1

- Now we will apply Strassen's Algorithm to a less optimal case. Let A be a (6×6) matrix, and B be a (6×6) matrix, and $C = S(A, B)$.
- The first problem with this matrix product is that A and B have sizes that are not powers of 2.

Strassen's Multiplication Example 1

- Now we will apply Strassen's Algorithm to a less optimal case. Let A be a (6×6) matrix, and B be a (6×6) matrix, and $C = S(A, B)$.
- The first problem with this matrix product is that A and B have sizes that are not powers of 2.
- We can account for this by altering our Strassen multiplication algorithm by including a stopping point for the recursion that is not the base case.
- What this means is that we will have the algorithm recurse once, giving us 7 more Strassen products to calculate. Each of these products will be (3×3) matrices.

Strassen's Multiplication Example 1

- Now we will apply Strassen's Algorithm to a less optimal case. Let A be a (6×6) matrix, and B be a (6×6) matrix, and $C = S(A, B)$.
- The first problem with this matrix product is that A and B have sizes that are not powers of 2.
- We can account for this by altering our Strassen multiplication algorithm by including a stopping point for the recursion that is not the base case.
- What this means is that we will have the algorithm recurse once, giving us 7 more Strassen products to calculate. Each of these products will be (3×3) matrices.
- At this point in the naive Strassen Algorithm, we could not divide a (3×3) matrix in half anymore, so we could not calculate the Strassen product. So we could not calculate our second level of with Strassen Multiplication. Our solution is to calculate the P_i with standard matrix multiplication.

Strassen's Multiplication Example 2

- Now we will apply Strassen's Algorithm to two rectangular matrices. Let A be a (47×32) matrix, and B be a (32×100) matrix. Let $C = S(A, B)$.
- The first problem with this matrix product is that both A and B are non square, and both A has a dimension that is not divisible by 2.

Strassen's Multiplication Example 2

- Now we will apply Strassen's Algorithm to two rectangular matrices. Let A be a (47×32) matrix, and B be a (32×100) matrix. Let $C = S(A, B)$.
- The first problem with this matrix product is that both A and B are non square, and both A has a dimension that is not divisible by 2.
- To use Strassen's matrix multiplication with this problem, we must alter the dimensions of the matrices slightly. So let A' be the matrix A augmented with a zero vector for its last row, so A' is a (48×32) matrix.
- We claim that $M(A', B)$ is equal to $M(A, B)$, but with a zero vector for its last row. Then we can compute $S(A', B)$ to find $S(A, B)$

Strassen's Multiplication Example 2 (continued)

- Finally, to compute the value of $S(A', B)$, we will need to use the pseudo-Strassen algorithm from example 1.
- The dimensions of the matrices are $\dim(A') = (48 \times 32)$, and $\dim(B) = (32 \times 100)$

Strassen's Multiplication Example 2 (continued)

- Finally, to compute the value of $S(A', B)$, we will need to use the pseudo-Strassen algorithm from example 1.
- The dimensions of the matrices are $\dim(A') = (48 \times 32)$, and $\dim(B) = (32 \times 100)$
- we can apply our Strassen algorithm for two levels of recursion, at which point the submatrices of A' will be (12×8), and the submatrices of B will be (8×25).
- Rather than compute the Strassen product of these matrices we can proceed with standard matrix multiplication.

Algorithm Efficiency

- Finally we will briefly discuss the efficiency of both standard matrix multiplication and Strassen's algorithm.
- To measure the efficiency of an algorithm, count the number of "basic operations" such as scalar addition, and scalar multiplication. The efficiency of the algorithm is a function of the size of its input.

Algorithm Efficiency

- Finally we will briefly discuss the efficiency of both standard matrix multiplication and Strassen's algorithm.
- To measure the efficiency of an algorithm, count the number of "basic operations" such as scalar addition, and scalar multiplication. The efficiency of the algorithm is a function of the size of its input.
- For the purpose of multiplying $(n \times n)$ matrices, the input of this function will be the sizes of the matrix, and the output is the number of multiplications.

Algorithm Efficiency

- Finally we will briefly discuss the efficiency of both standard matrix multiplication and Strassen's algorithm.
- To measure the efficiency of an algorithm, count the number of "basic operations" such as scalar addition, and scalar multiplication. The efficiency of the algorithm is a function of the size of its input.
- For the purpose of multiplying $(n \times n)$ matrices, the input of this function will be the sizes of the matrix, and the output is the number of multiplications.
- Finally, an algorithm is said to have an order of growth, which is an upper bound for the growth rate.
- This bound is commonly known as "Big-O" notation in math and computer science.

Matrix Multiplication Efficiency

- First we will look at the efficiency of standard matrix multiplication.
- for a matrix $M(A, B)$, The number of multiplications for each entry $[AB]_{i,j}$ is given by:

$$\begin{aligned}[AB]_{i,j} &= [A]_{i,1}[B]_{1,j} + [A]_{i,2}[B]_{2,j} + \ldots + [A]_{i,n}[B]_{n,j} \\ &= \sum_{k=1}^{n} [A]_{i,k}[B]_{k,j}\end{aligned}$$

Matrix Multiplication Efficiency

- First we will look at the efficiency of standard matrix multiplication.
- for a matrix $M(A, B)$, The number of multiplications for each entry $[AB]_{i,j}$ is given by:

$$\begin{aligned}[AB]_{i,j} &= [A]_{i,1}[B]_{1,j} + [A]_{i,2}[B]_{2,j} + \ldots + [A]_{i,n}[B]_{n,j} \\ &= \sum_{k=1}^{n} [A]_{i,k}[B]_{k,j}\end{aligned}$$

- So entry entry takes n multiplication operations to calculate.
- Because each n of these calculations must take place n times for each column and n times for each row, standard matrix multiplication takes $n \cdot n \cdot n = n^3$ multiplications, which means it is an algorithm with an efficiency of $O(n^3)$.

Strassen Algorithm Efficiency

- Next we will look at the efficiency of Strassen's algorithm. Let A and B be $(n \times n)$ matrices, with $n = 2^q$.
- Because Strassen is a recursive algorithm, we count the number of operations that it takes to calculate the entire product $S(A, B)$.

Strassen Algorithm Efficiency

- Next we will look at the efficiency of Strassen's algorithm. Let A and B be $(n \times n)$ matrices, with $n = 2^q$.
- Because Strassen is a recursive algorithm, we count the number of operations that it takes to calculate the entire product $S(A, B)$.
- In our definition of Strassen's algorithm, when we calculate the product $S(A, B)$, we must recurse and calculate 7 more Strassen products, one for each $\hat{P}_i$
- Each of these Strassen products take matrices that have **half** the length of its parent matrix.

Strassen Algorithm Efficiency

- Next we will look at the efficiency of Strassen's algorithm. Let A and B be $(n \times n)$ matrices, with $n = 2^q$.
- Because Strassen is a recursive algorithm, we count the number of operations that it takes to calculate the entire product $S(A, B)$.
- In our definition of Strassen's algorithm, when we calculate the product $S(A, B)$, we must recurse and calculate 7 more Strassen products, one for each $\hat{P}_i$
- Each of these Strassen products take matrices that have **half** the length of its parent matrix.
- In our example, $S(A, B)$ takes matrices of size $(2^q \times 2^q)$. To solve it it will need to calculate a Strassen product of 7 more matrices of size $(2^{q-1} \times 2^{q-1})$.

Strassen Algorithm Efficiency (continued)

- As we saw earlier, the number of levels of recursion that Strassen's algorithm will have for a $(2^q \times 2^q)$ matrix is $\log_2 2^q = q$.
- Each of these q levels have 7 more branches of recursion, so the order of operations of Strassen's algorithm is $O(7^q)$.

Strassen Algorithm Efficiency (continued)

- As we saw earlier, the number of levels of recursion that Strassen's algorithm will have for a $(2^q \times 2^q)$ matrix is $\log_2 2^q = q$.
- Each of these q levels have 7 more branches of recursion, so the order of operations of Strassen's algorithm is $O(7^q)$.
- However, $q = \log_2 2^q = \log_2 n$, and $7^{\log_2 n} = n^{\log_2 7}$, which is approximately $n^{2.807}$.
- So Strassen's Algorithm is an $O(n^{\log_2 7})$ algorithm, meaning it has a slightly lower order of growth than standard matrix multiplication.

Strassen Algorithm Efficiency (continued)

- The efficiency gains from Strassen's algorithm can be very beneficial given the correct circumstances. The number of operations that can be reduced is almost $n^3 - n^{\log_2 7}$.
- However, there are many drawbacks to Strassen's algorithm.

Strassen Algorithm Efficiency (continued)

- The efficiency gains from Strassen's algorithm can be very beneficial given the correct circumstances. The number of operations that can be reduced is almost $n^3 - n^{\log_2 7}$.
- However, there are many drawbacks to Strassen's algorithm.
- The largest issue with the algorithm is that it is not always usable. Matrices that are "very rectangular" work poorly with the algorithm, and matrix vector products cannot be calculated with it.
- The second issue with the algorithm is that it has a large overhead of calculation costs. For smaller matrices, calculating $\hat{P}_i$ and $\hat{C}_{i,j}$ is often more costly than standard matrix multiplication.
- In fact, Strassen's algorithm generally does not have efficiency gains over standard matrix multiplication unless the size of the matrices are about (100×100) or larger [3].

Conclusion

- Strassen's algorithm is an interesting alternative to standard matrix multiplication.
- Strassen has an asymptotic growth of $O(n^{log_2 7})$ while standard matrix multiplication has an asymptotic growth of $O(n^3)$.

Conclusion

- Strassen's algorithm is an interesting alternative to standard matrix multiplication.
- Strassen has an asymptotic growth of $O(n^{log_2 7})$ while standard matrix multiplication has an asymptotic growth of $O(n^3)$.
- The algorithm is only usable when on matrices that have sizes divisible by 2.
- There is a large overhead cost of addition operations that make the algorithm inefficient for small matrix multiplication.

Sources

[1] Beezer, Robert A. A First Course in Linear Algebra. Tacoma, Wash: Beezer, 2008. Print.

[2] Golub, Gene H, and Loan C. F. Van. Matrix Computations. Baltimore: Johns Hopkins University Press, 1983. Print.

[3] Bunch, James R, and John E. Hopcroft. "Triangular Factorization and Inversion by Fast Matrix Multiplication." Mathematics of Computation. 28.125 (1974): 231. Print.