Fast Matrix Multiplication

Michael Schmid

31.05.2021

Big $\mathcal{O}$ notation

- Time complexity of an algorithm

Big $\mathcal{O}$ notation

- Time complexity of an algorithm
- How many multiplications in a function

Big $\mathcal{O}$ notation

- Time complexity of an algorithm
- How many multiplications in a function
- Drop Constants

Big $\mathcal{O}$ notation

Algorithm 1 Square Matrix Multiplication

function FOO(a, b)

    return $a + b$

Big $\mathcal{O}$ notation

Algorithm 2 Square Matrix Multiplication

```
function FOO(a, b)
    return a + b
```

$\mathcal{O}(1)$

Big $\mathcal{O}$ notation

Algorithm 3 Square Matrix Multiplication

```
function FOO(a, b)
    x ← a + b
    y ← a · b
    return x + y
```

Big $\mathcal{O}$ notation

Algorithm 4 Square Matrix Multiplication

```
function FOO(a, b)
    x ← a + b
    y ← a · b
    return x + y
```

$$\mathcal{O}(1) + \mathcal{O}(1) = 2\mathcal{O}(1) = \mathcal{O}(1)$$

Big $\mathcal{O}$ notation

Algorithm 5 Square Matrix Multiplication

---

function $\textsc{Foo}(\mathbf{A}, \mathbf{B}, \mathrm{n})$
    $sum \leftarrow 0$
    for $i = 0, 1, 2 \ldots, n$ **do**
        $sum \leftarrow sum + A[i] \cdot B[i]$
    return sum

Big $\mathcal{O}$ notation

Algorithm 6 Square Matrix Multiplication

function FOO(**A**, **B**,n)
  $sum \leftarrow 0$
  for $i = 0, 1, 2 \ldots, n$ **do**
    $sum \leftarrow sum + A[i] \cdot B[i]$
  return sum

$\mathcal{O}(n)$

Big $\mathcal{O}$ notation

Algorithm 7 Square Matrix Multiplication

function FOO(**A**, **B**,n)
    $sum \leftarrow 0$
    for $i = 0, 1, 2 \ldots, n$ **do**
        for $j = 0, 1, 2 \ldots, n$ **do**
            $sum \leftarrow sum + A[i] \cdot B[j]$
    return sum

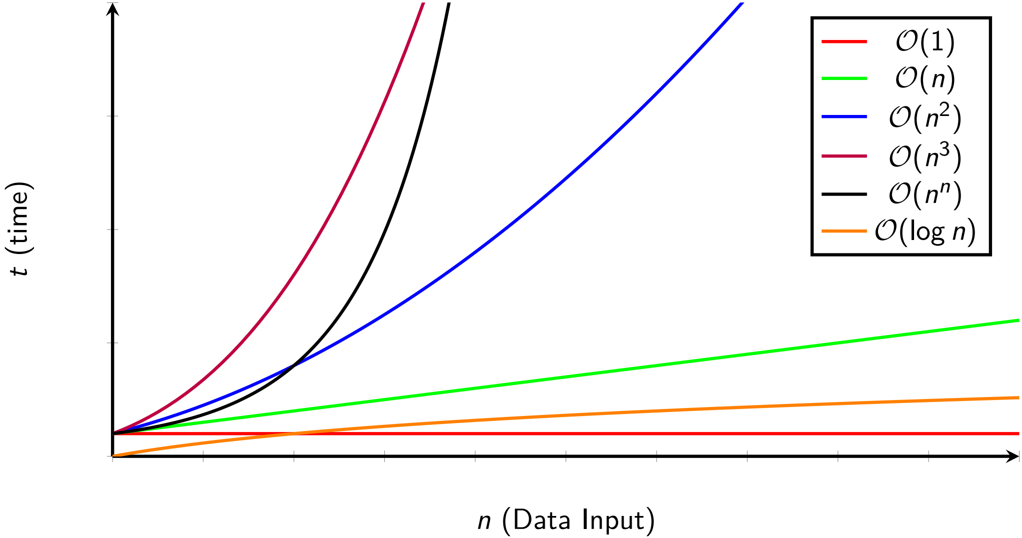
Big $\mathcal{O}$ notation

Algorithm 8 Square Matrix Multiplication

function FOO(**A**, **B**,n)
    $sum \leftarrow 0$
    for $i = 0, 1, 2 \ldots, n$ **do**
        for $j = 0, 1, 2 \ldots, n$ **do**
            $sum \leftarrow sum + A[i] \cdot B[j]$
    return sum

$\mathcal{O}(n^2)$

Big $\mathcal{O}$ notation



Strassen's Algorithm

$$\mathbf{AB} = \mathbf{C}$$

Strassen's Algorithm

$$\mathbf{AB} = \mathbf{C}$$

$$\begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \begin{bmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{bmatrix} = \begin{bmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{bmatrix}$$

Strassen's Algorithm

$$\mathbf{AB} = \mathbf{C}$$

$$\begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \begin{bmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{bmatrix} = \begin{bmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{bmatrix}$$

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

Algorithm

Algorithm 9 Square Matrix Multiplication

function $\mathrm{MM}(\mathbf{A}, \mathbf{B}, \mathbf{C})$
    $sum \leftarrow 0$
    $n \leftarrow columns(\mathbf{A}) == rows(\mathbf{B})$
    $m \leftarrow rows(\mathbf{A})$
    $p \leftarrow columns(\mathbf{B})$
    for $i = 0, 1, 2 \ldots, m-1$ **do**
        for $j = 0, 1, 2 \ldots, p-1$ **do**
            $sum \leftarrow 0$
            for $k = 0, 1, 2 \ldots, n-1$ **do**
                $sum \leftarrow sum + \mathbf{A}[i][k] \cdot \mathbf{B}[k][j]$
            $\mathbf{C}[i][j] \leftarrow sum$
    return C

$$\begin{array}{cc} & \begin{bmatrix} B_{1,1} & \cdots & B_{1,j} & \cdots & B_{1,p} \\ \vdots & & \vdots & & \vdots \\ B_{k,1} & \cdots & B_{k,j} & \cdots & B_{k,p} \\ \vdots & & \vdots & & \vdots \\ B_{n,1} & \cdots & B_{n,j} & \cdots & B_{n,p} \end{bmatrix} \\ \begin{bmatrix} A_{1,1} & \cdots & A_{1,k} & \cdots & A_{1,n} \\ \vdots & & \vdots & & \vdots \\ A_{i,1} & \cdots & A_{i,k} & \cdots & A_{i,n} \\ \vdots & & \vdots & & \vdots \\ A_{m,1} & \cdots & A_{m,k} & \cdots & A_{m,n} \end{bmatrix} & \begin{bmatrix} C_{1,1} & \cdots & C_{1,j} & \cdots & C_{1,p} \\ \vdots & & \vdots & & \vdots \\ C_{i,1} & \cdots & C_{i,j} & \cdots & C_{i,p} \\ \vdots & & \vdots & & \vdots \\ C_{m,1} & \cdots & C_{m,j} & \cdots & C_{m,p} \end{bmatrix} \end{array}$$

Algorithm

$$\begin{bmatrix} B_{1,1} & \cdots & B_{1,j} & \cdots & B_{1,p} \\ \vdots & & \vdots & & \vdots \\ B_{k,1} & \cdots & B_{k,j} & \cdots & B_{k,p} \\ \vdots & & \vdots & & \vdots \\ B_{n,1} & \cdots & B_{n,j} & \cdots & B_{n,p} \end{bmatrix}$$

$$\begin{bmatrix} A_{1,1} & \cdots & A_{1,k} & \cdots & A_{1,n} \\ \vdots & & \vdots & & \vdots \\ A_{i,1} & \cdots & A_{i,k} & \cdots & A_{i,n} \\ \vdots & & \vdots & & \vdots \\ A_{m,1} & \cdots & A_{m,k} & \cdots & A_{m,n} \end{bmatrix} \begin{bmatrix} C_{1,1} & \cdots & C_{1,j} & \cdots & C_{1,p} \\ \vdots & & \vdots & & \vdots \\ C_{i,1} & \cdots & C_{i,j} & \cdots & C_{i,p} \\ \vdots & & \vdots & & \vdots \\ C_{m,1} & \cdots & C_{m,j} & \cdots & C_{m,p} \end{bmatrix}$$

Strassen's Algorithm

$$
\begin{aligned}
\mathrm{I} &= (A_{11} + A_{22}) \cdot (B_{11} + B_{22}) \\
\mathrm{II} &= (A_{21} + A_{22}) \cdot B_{11} \\
\mathrm{III} &= A_{11} \cdot (B_{12} - B_{22}) \\
\mathrm{IV} &= A_{22} \cdot (-B_{11} + B_{21}) \\
\mathrm{V} &= (A_{11} + A_{12}) \cdot B_{22} \\
\mathrm{VI} &= (-A_{11} + A_{21}) \cdot (B_{11} + B_{12}) \\
\mathrm{VII} &= (A_{12} - A_{22}) \cdot (B_{21} + B_{22})
\end{aligned}
$$

Strassen's Algorithm

$$
\begin{aligned}
\mathrm{I} &= (A_{11} + A_{22}) \cdot (B_{11} + B_{22}) \\
\mathrm{II} &= (A_{21} + A_{22}) \cdot B_{11} \\
\mathrm{III} &= A_{11} \cdot (B_{12} - B_{22}) \\
\mathrm{IV} &= A_{22} \cdot (-B_{11} + B_{21}) \\
\mathrm{V} &= (A_{11} + A_{12}) \cdot B_{22} \\
\mathrm{VI} &= (-A_{11} + A_{21}) \cdot (B_{11} + B_{12}) \\
\mathrm{VII} &= (A_{12} - A_{22}) \cdot (B_{21} + B_{22})
\end{aligned}
$$

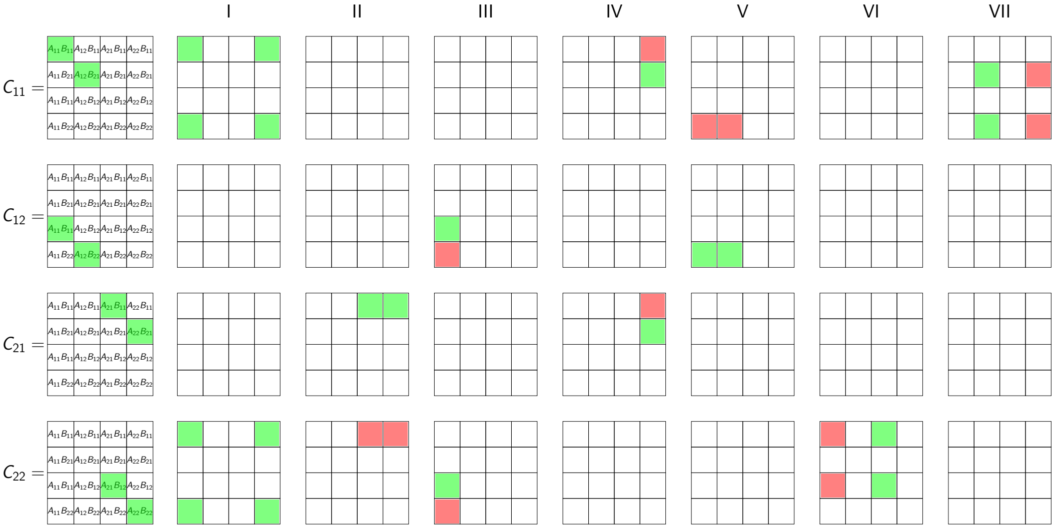
$$
\begin{aligned}
C_{11} &= \mathrm{I} + \mathrm{IV} - \mathrm{V} + \mathrm{VII} \\
C_{21} &= \mathrm{II} + \mathrm{IV} \\
C_{12} &= \mathrm{III} + \mathrm{V} \\
C_{22} &= \mathrm{I} + \mathrm{III} - \mathrm{II} + \mathrm{VI}
\end{aligned}
$$

Strassen's Algorithm

$$
\begin{aligned}
\mathrm{I} &= (A_{11} + A_{22}) \cdot (B_{11} + B_{22}) \\
\mathrm{II} &= (A_{21} + A_{22}) \cdot B_{11} \\
\mathrm{III} &= A_{11} \cdot (B_{12} - B_{22}) \\
\mathrm{IV} &= A_{22} \cdot (-B_{11} + B_{21}) \\
\mathrm{V} &= (A_{11} + A_{12}) \cdot B_{22} \\
\mathrm{VI} &= (-A_{11} + A_{21}) \cdot (B_{11} + B_{12}) \\
\mathrm{VII} &= (A_{12} - A_{22}) \cdot (B_{21} + B_{22})
\end{aligned}
$$

$$
\begin{aligned}
C_{11} &= \mathrm{I} + \mathrm{IV} - \mathrm{V} + \mathrm{VII} \\
C_{21} &= \mathrm{II} + \mathrm{IV} \\
C_{12} &= \mathrm{III} + \mathrm{V} \\
C_{22} &= \mathrm{I} + \mathrm{III} - \mathrm{II} + \mathrm{VI}
\end{aligned}
$$

$$
\begin{aligned}
C_{11} &= (A_{11} + A_{22}) \cdot (B_{11} + B_{22}) + A_{22} \cdot (-B_{11} + B_{21}) - (A_{11} + A_{12}) \cdot B_{22} + (A_{12} - A_{22}) \cdot (B_{21} + B_{22}) \\
C_{11} &= A_{11}B_{11} + A_{11}B_{22} + A_{22}B_{11} + A_{22}B_{22} - A_{22}B_{11} + A_{22}B_{21} - A_{11}B_{22} - A_{12}B_{22} + A_{12}B_{21} + A_{12}B_{22} - A_{22}B_{21} - A_{22}B_{22} \\
C_{11} &= A_{11}B_{11} + A_{12}B_{21}
\end{aligned}
$$


I
II
III
IV
V
VI
VII
$C_{11} =$
$C_{12} =$
$C_{21} =$
$C_{22} =$
$A_{11}B_{11}$ $A_{12}B_{11}$ $A_{21}B_{11}$ $A_{22}B_{11}$
$A_{11}B_{21}$ $A_{12}B_{21}$ $A_{21}B_{21}$ $A_{22}B_{21}$
$A_{11}B_{11}$ $A_{12}B_{12}$ $A_{21}B_{12}$ $A_{22}B_{12}$
$A_{11}B_{22}$ $A_{12}B_{22}$ $A_{21}B_{22}$ $A_{22}B_{22}$

Strassen's Algorithm

$$\begin{aligned}
\mathrm{I} &= (A_{11} + A_{22}) \cdot (B_{11} + B_{22}) \\
\mathrm{II} &= (A_{21} + A_{22}) \cdot B_{11} \\
\mathrm{III} &= A_{11} \cdot (B_{12} - B_{22}) \\
\mathrm{IV} &= A_{22} \cdot (-B_{11} + B_{21}) \\
\mathrm{V} &= (A_{11} + A_{12}) \cdot B_{22} \\
\mathrm{VI} &= (-A_{11} + A_{21}) \cdot (B_{11} + B_{12}) \\
\mathrm{VII} &= (A_{12} - A_{22}) \cdot (B_{21} + B_{22})
\end{aligned}$$

$$\begin{aligned}
C_{11} &= \mathrm{I} + \mathrm{IV} - \mathrm{V} + \mathrm{VII} \\
C_{21} &= \mathrm{II} + \mathrm{IV} \\
C_{12} &= \mathrm{III} + \mathrm{V} \\
C_{22} &= \mathrm{I} + \mathrm{III} - \mathrm{II} + \mathrm{VI}
\end{aligned}$$

$$
\begin{aligned}
\mathbf{I} &= (\mathbf{A}_{11} + \mathbf{A}_{22}) \cdot (\mathbf{B}_{11} + \mathbf{B}_{22}) \\
\mathbf{II} &= (\mathbf{A}_{21} + \mathbf{A}_{22}) \cdot \mathbf{B}_{11} \\
\mathbf{III} &= \mathbf{A}_{11} \cdot (\mathbf{B}_{12} - \mathbf{B}_{22}) \\
\mathbf{IV} &= \mathbf{A}_{22} \cdot (-\mathbf{B}_{11} + \mathbf{B}_{21}) \\
\mathbf{V} &= (\mathbf{A}_{11} + \mathbf{A}_{12}) \cdot \mathbf{B}_{22} \\
\mathbf{VI} &= (-\mathbf{A}_{11} + \mathbf{A}_{21}) \cdot (\mathbf{B}_{11} + \mathbf{B}_{12}) \\
\mathbf{VII} &= (\mathbf{A}_{12} - \mathbf{A}_{22}) \cdot (\mathbf{B}_{21} + \mathbf{B}_{22})
\end{aligned}
$$

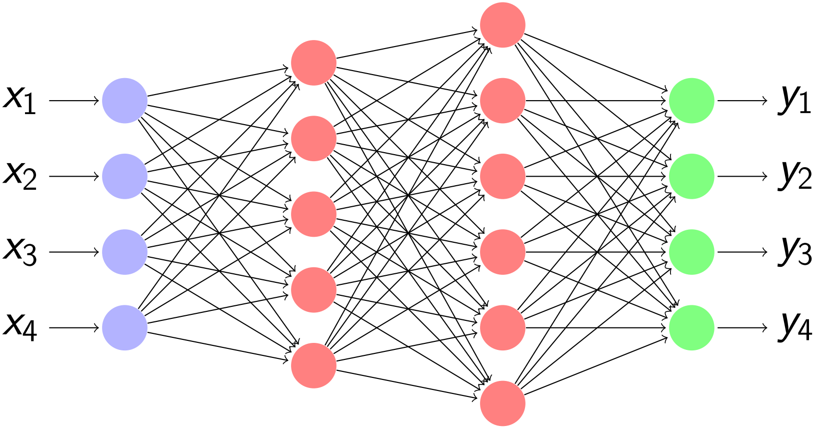
$$
\begin{aligned}
\mathbf{C}_{11} &= \mathbf{I} + \mathbf{IV} - \mathbf{V} + \mathbf{VII} \\
\mathbf{C}_{21} &= \mathbf{II} + \mathbf{IV} \\
\mathbf{C}_{12} &= \mathbf{III} + \mathbf{V} \\
\mathbf{C}_{22} &= \mathbf{I} + \mathbf{III} - \mathbf{II} + \mathbf{VI}
\end{aligned}
$$

Algorithm

Algorithm 10 Strassen Matrix Multiplication

```
function MM(A, B, C, n)
    if n = 2 then
        C ← zeros((n, n))
        P ← (A[0][0] + A[1][1]) · (B[0][0] + B[1][1])
        Q ← (A[1][0] + A[1][1]) · B[0][0]
        R ← A[0][0] · (B[0][1] − B[1][1])
        S ← A[1][1] · (B[1][0] − B[0][0])
        T ← (A[0][0] + A[0][1]) · B[1][1]
        U ← (A[1][0] − A[0][0]) · (B[0][0] + B[0][1])
        V ← (A[0][1] − A[1][1]) · (B[1][0] + B[1][1])
        C[0][0] ← P + S − T + V
        C[0][1] ← R + T
        C[1][0] ← Q + S
        C[1][1] ← P + R − Q + U
    else
        m ← n/2
        A11, A12, A21, A22 ← A[: m][: m], A[: m][m :], A[m :][: m], A[m :][m :]
        B11, B12, B21, B22 ← B[: m][: m], B[: m][m :], B[m :][: m], B[m :][m :]
        C11 ← strassen(A11, B11, m) + strassen(A12, B21, m)
        C12 ← strassen(A11, B12, m) + strassen(A12, B22, m)
        C21 ← strassen(A21, B11, m) + strassen(A22, B21, m)
        C22 ← strassen(A21, B12, m) + strassen(A22, B22, m)
        C ← vstack((hstack((C11, C12)), hstack((C21, C22))))
    return C
```

Neural Network



Vector-Matrix Multiplication

$$\begin{bmatrix} A_{1,1} & A_{1,2} & A_{1,3} & A_{1,4} \end{bmatrix} \begin{bmatrix} B_{1,1} & B_{1,2} & B_{1,3} & B_{1,4} & B_{1,5} \\ B_{2,1} & B_{2,2} & B_{2,3} & B_{2,4} & B_{2,5} \\ B_{3,1} & B_{3,2} & B_{3,3} & B_{3,4} & B_{3,5} \\ B_{4,1} & B_{4,2} & B_{4,3} & B_{4,4} & B_{4,5} \end{bmatrix}$$

$$\begin{bmatrix} C_{1,1} & C_{1,2} & C_{1,3} & C_{1,4} & C_{1,5} \end{bmatrix}$$

Vector-Matrix Multiplication

$$\begin{bmatrix} A_{1,1} & A_{1,2} & A_{1,3} & A_{1,4} \end{bmatrix} \begin{bmatrix} B_{1,1} & B_{1,2} & B_{1,3} & B_{1,4} & B_{1,5} \\ B_{2,1} & B_{2,2} & B_{2,3} & B_{2,4} & B_{2,5} \\ B_{3,1} & B_{3,2} & B_{3,3} & B_{3,4} & B_{3,5} \\ B_{4,1} & B_{4,2} & B_{4,3} & B_{4,4} & B_{4,5} \end{bmatrix}$$

$$\begin{bmatrix} C_{1,1} & C_{1,2} & C_{1,3} & C_{1,4} & C_{1,5} \end{bmatrix}$$

Vector-Matrix Multiplication

$$\begin{bmatrix} A_{1,1} & A_{1,2} & A_{1,3} & A_{1,4} \end{bmatrix} \begin{bmatrix} B_{1,1} & B_{1,2} & B_{1,3} & B_{1,4} & B_{1,5} \\ B_{2,1} & B_{2,2} & B_{2,3} & B_{2,4} & B_{2,5} \\ B_{3,1} & B_{3,2} & B_{3,3} & B_{3,4} & B_{3,5} \\ B_{4,1} & B_{4,2} & B_{4,3} & B_{4,4} & B_{4,5} \end{bmatrix}$$

$$\begin{bmatrix} C_{1,1} & C_{1,2} & C_{1,3} & C_{1,4} & C_{1,5} \end{bmatrix}$$

Vector-Matrix Multiplication

$$\begin{bmatrix} A_{1,1} & A_{1,2} & A_{1,3} & A_{1,4} \end{bmatrix} \begin{bmatrix} B_{1,1} & B_{1,2} & B_{1,3} & B_{1,4} & B_{1,5} \\ B_{2,1} & B_{2,2} & B_{2,3} & B_{2,4} & B_{2,5} \\ B_{3,1} & B_{3,2} & B_{3,3} & B_{3,4} & B_{3,5} \\ B_{4,1} & B_{4,2} & B_{4,3} & B_{4,4} & B_{4,5} \end{bmatrix}$$

$$\begin{bmatrix} C_{1,1} & C_{1,2} & C_{1,3} & C_{1,4} & C_{1,5} \end{bmatrix}$$

Vector-Matrix Multiplication

$$\begin{bmatrix} A_{1,1} & A_{1,2} & A_{1,3} & A_{1,4} \end{bmatrix} \begin{bmatrix} B_{1,1} & B_{1,2} & B_{1,3} & B_{1,4} & B_{1,5} \\ B_{2,1} & B_{2,2} & B_{2,3} & B_{2,4} & B_{2,5} \\ B_{3,1} & B_{3,2} & B_{3,3} & B_{3,4} & B_{3,5} \\ B_{4,1} & B_{4,2} & B_{4,3} & B_{4,4} & B_{4,5} \end{bmatrix}$$

$$\begin{bmatrix} C_{1,1} & C_{1,2} & C_{1,3} & C_{1,4} & C_{1,5} \end{bmatrix}$$

DSP Architecture