Divide-and-Conquer algorithsm for matrix multiplication

$$A = \begin{pmatrix} A^{11} & A^{12} \\ A^{21} & A^{22} \end{pmatrix} \quad B = \begin{pmatrix} B^{11} & B^{12} \\ B^{21} & B^{22} \end{pmatrix} \quad C = A \times B = \begin{pmatrix} C^{11} & C^{12} \\ C^{21} & C^{22} \end{pmatrix}$$

Formulas for $C^{11}, C^{12}, C^{21}, C^{22}$:

$$C^{11} = A^{11}B^{11} + A^{12}B^{21} \qquad C^{12} = A^{11}B^{12} + A^{12}B^{22}$$
$$C^{21} = A^{21}B^{11} + A^{22}B^{21} \qquad C^{22} = A^{21}B^{12} + A^{22}B^{22}$$

The First Attempt Straightforward from the formulas above (assuming that n is a power of 2):

MMult(A, B, n)

1. If $n = 1$ Output $A \times B$
2. Else
3. Compute $A^{11}, B^{11}, \ldots, A^{22}, B^{22}$ % by computing $m = n/2$
4. $X_1 \leftarrow MMult(A^{11}, B^{11}, n/2)$
5. $X_2 \leftarrow MMult(A^{12}, B^{21}, n/2)$
6. $X_3 \leftarrow MMult(A^{11}, B^{12}, n/2)$
7. $X_4 \leftarrow MMult(A^{12}, B^{22}, n/2)$
8. $X_5 \leftarrow MMult(A^{21}, B^{11}, n/2)$
9. $X_6 \leftarrow MMult(A^{22}, B^{21}, n/2)$
10. $X_7 \leftarrow MMult(A^{21}, B^{12}, n/2)$
11. $X_8 \leftarrow MMult(A^{22}, B^{22}, n/2)$
12. $C^{11} \leftarrow X_1 + X_2$
13. $C^{12} \leftarrow X_3 + X_4$
14. $C^{21} \leftarrow X_5 + X_6$
15. $C^{22} \leftarrow X_7 + X_8$
16. Output C
17. End If

Analysis: The operations on line 3 take constant time. The combining cost (lines 12–15) is $\Theta(n^2)$ (adding two $\frac{n}{2} \times \frac{n}{2}$ matrices takes time $\frac{n^2}{4} = \Theta(n^2)$). There are 8 recursive calls (lines 4–11). So let $T(n)$ be the total number of mathematical operations performed by MMult(A, B, n), then

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

The Master Theorem gives us

$$T(n) = \Theta(n^{\log_2(8)}) = \Theta(n^3)$$

So this is not an improvement on the "obvious" algorithm given earlier (that uses n^3 operations).

Strassen's algorithm is based on the following observation:

$$C^{11} = P_5 + P_4 - P_2 + P_6 \qquad C^{12} = P_1 + P_2$$
$$C^{21} = P_3 + P_4 \qquad C^{22} = P_1 + P_5 - P_3 - P_7$$

where

$$\begin{aligned}
P_1 &= A^{11}(B^{12} - B^{22}) \\
P_2 &= (A^{11} + A^{12})B^{22} \\
P_3 &= (A^{21} + A^{22})B^{11} \\
P_4 &= A^{22}(B^{21} - B^{11}) \\
P_5 &= (A^{11} + A^{22})(B^{11} + B^{22}) \\
P_6 &= (A^{12} - A^{22})(B^{21} + B^{22}) \\
P_7 &= (A^{11} - A^{21})(B^{11} + B^{12})
\end{aligned}$$

Exercise Verify that $C^{11}, \ldots, C^{22}$ can be computed as above.

The above formulas can be used to compute $A \times B$ recursively as follows:

Strassen(A, B)

1. If $n = 1$ Output $A \times B$
2. Else
3. Compute $A^{11}, B^{11}, \ldots, A^{22}, B^{22}$ % by computing $m = n/2$
4. $P_1 \leftarrow Strassen(A^{11}, B^{12} - B^{22})$
5. $P_2 \leftarrow Strassen(A^{11} + A^{12}, B^{22})$
6. $P_3 \leftarrow Strassen(A^{21} + A^{22}, B^{11})$
7. $P_4 \leftarrow Strassen(A^{22}, B^{21} - B^{11})$
8. $P_5 \leftarrow Strassen(A^{11} + A^{22}, B^{11} + B^{22})$
9. $P_6 \leftarrow Strassen(A^{12} - A^{22}, B^{21} + B^{22})$
10. $P_7 \leftarrow Strassen(A^{11} - A^{21}, B^{11} + B^{12})$
11. $C^{11} \leftarrow P_5 + P_4 - P_2 + P_6$
12. $C^{12} \leftarrow P_1 + P_2$
13. $C^{21} \leftarrow P_3 + P_4$
14. $C^{22} \leftarrow P_1 + P_5 - P_3 - P_7$
15. Output C
16. End If

Analysis: The operations on line 3 take constant time. The combining cost (lines 11–14) is $\Theta(n^2)$. There are 7 recursive calls (lines 4–10). So let $T(n)$ be the total number of mathematical operations performed by Strassen(A, B), then

$$T(n) = 7T(\frac{n}{2}) + \Theta(n^2)$$

The Master Theorem gives us

$$T(n) = \Theta(n^{\log_2(7)}) = \Theta(n^{2.8})$$

The best current upper bound for multiplying two matrices of size $n \times n$ is $\mathcal{O}(n^{2.32})$ (by using similar idea, but instead of dividing a matrix into 4 quaters, people divide them into a bigger number of submatrices).

Open question: Can we multiply two $n \times n$ matrices in time $\mathcal{O}(n^2)$?