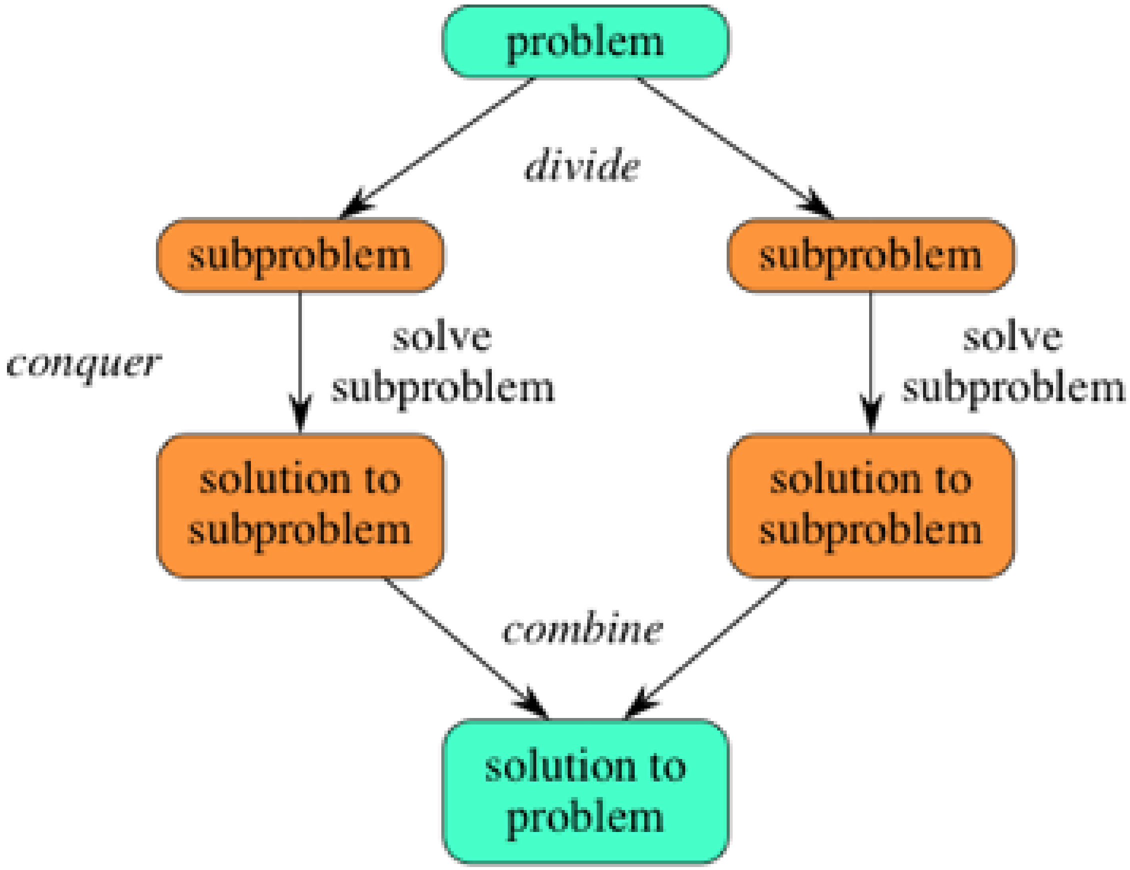
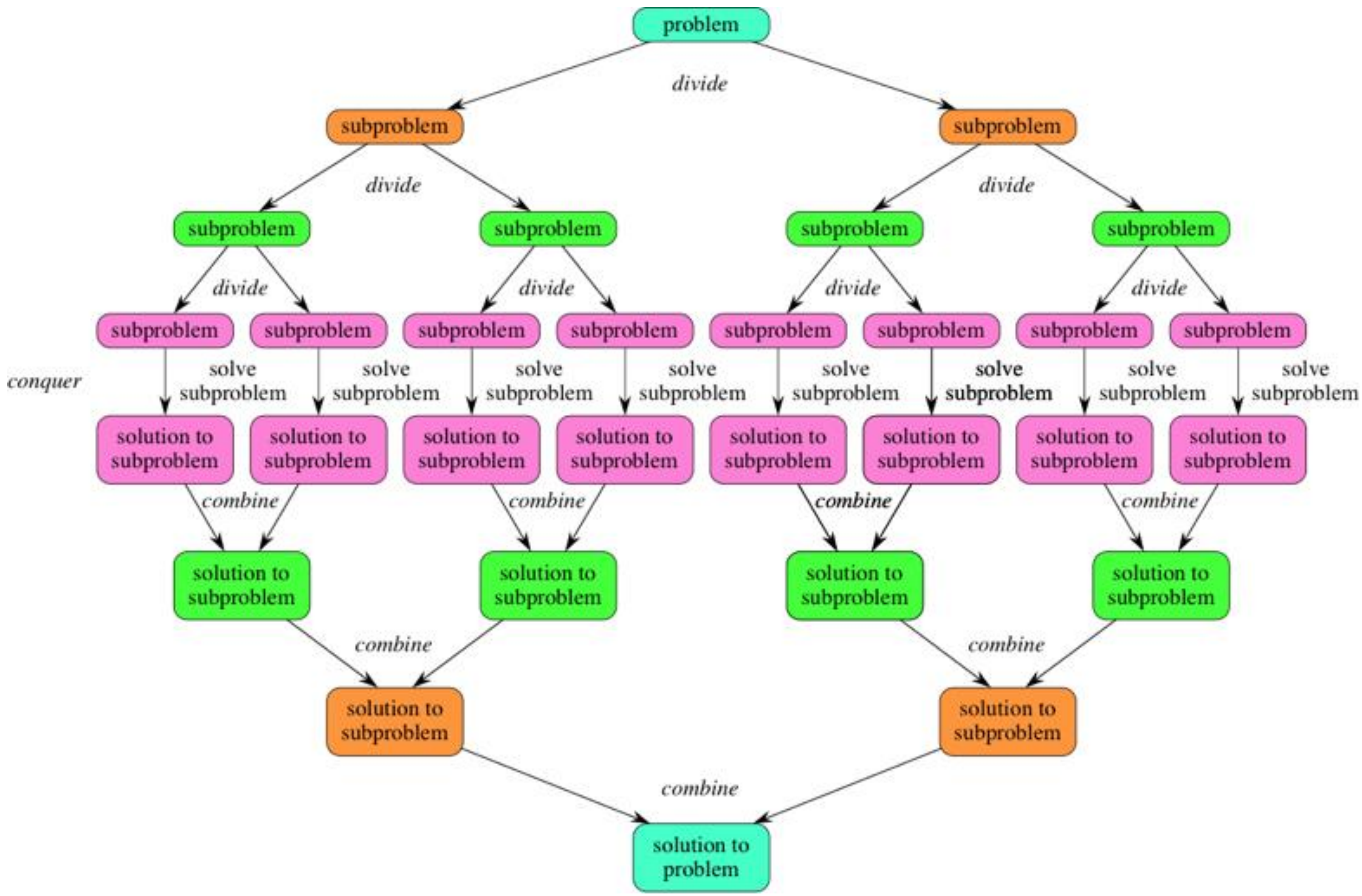
Divide Conquer and Combine Algorithm




problem
divide
subproblem
subproblem
divide
subproblem
subproblem
divide
subproblem
subproblem
subproblem
subproblem
divide
subproblem
subproblem
divide
divide
divide
subproblem
subproblem
subproblem
subproblem
subproblem
subproblem
subproblem
subproblem
conquer
solve subproblem
solve subproblem
solve subproblem
solve subproblem
solve subproblem
solve subproblem
solve subproblem
solve subproblem
solution to subproblem
solution to subproblem
solution to subproblem
solution to subproblem
solution to subproblem
solution to subproblem
solution to subproblem
solution to subproblem
combine
combine
combine
combine
solution to subproblem
solution to subproblem
solution to subproblem
solution to subproblem
combine
combine
solution to subproblem
solution to subproblem
combine
solution to problem

Matrix Multiplication

$$\begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix} \begin{bmatrix} j & k & l \\ m & n & o \\ p & q & r \end{bmatrix} = \begin{bmatrix} (aj + bm + cp) & (ak + bn + cq) & (al + bo + cr) \\ (dj + em + fp) & (dk + en + fq) & (dl + eo + fr) \\ (gj + hm + ip) & (gk + hn + iq) & (gl + ho + ir) \end{bmatrix}$$

$$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix} \times \begin{bmatrix} 10 & 11 \\ 20 & 21 \\ 30 & 31 \end{bmatrix}$$

$$= \begin{bmatrix} 1x10 + 2x20 + 3x30 & 1x11 + 2x21 + 3x31 \\ 4x10 + 5x20 + 6x30 & 4x11 + 5x21 + 6x31 \end{bmatrix}$$

$$= \begin{bmatrix} 10+40+90 & 11+42+93 \\ 40+100+180 & 44+105+186 \end{bmatrix} = \begin{bmatrix} 140 & 146 \\ 320 & 335 \end{bmatrix}$$

Matrix Multiplication

SQUARE-MATRIX-MULTIPLY(A, B)

```
n = A.rows
let C be a new n × n matrix
for i = 1 to n
    for j = 1 to n
        c_ij = 0
        for k = 1 to n
            c_ij = c_ij + a_ik · b_kj
return C
```

Cost = $O(n^3)$

Matrix Multiplication

Matrix multiplication. Given two n-by-n matrices A and B, compute $C = AB$.

Grade-school. $\Theta(n^3)$ arithmetic operations.

$$\begin{bmatrix} c_{11} & c_{12} & \cdots & c_{1n} \\ c_{21} & c_{22} & \cdots & c_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ c_{n1} & c_{n2} & \cdots & c_{nn} \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{bmatrix} \times \begin{bmatrix} b_{11} & b_{12} & \cdots & b_{1n} \\ b_{21} & b_{22} & \cdots & b_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ b_{n1} & b_{n2} & \cdots & b_{nn} \end{bmatrix}$$

$$\begin{bmatrix} .59 & .32 & .41 \\ .31 & .36 & .25 \\ .45 & .31 & .42 \end{bmatrix} = \begin{bmatrix} .70 & .20 & .10 \\ .30 & .60 & .10 \\ .50 & .10 & .40 \end{bmatrix} \times \begin{bmatrix} .80 & .30 & .50 \\ .10 & .40 & .10 \\ .10 & .30 & .40 \end{bmatrix}$$

Q. Can we perform matrix multiplication more effeciantly

Consider $A = \begin{pmatrix} 1 & 2 & 1 & 2 \\ 3 & 8 & 2 & 2 \\ 5 & 1 & 4 & 9 \\ 6 & 2 & 5 & 0 \end{pmatrix}$ and $B = \begin{pmatrix} 1 & 2 & 5 & 6 \\ 3 & 4 & 7 & 8 \\ 9 & 1 & 4 & 5 \\ 2 & 3 & 6 & 7 \end{pmatrix}$. I'll use MATRIX-MULTIPLY-RECURSIVE(MMR) algo to multiply A and B.

Since $n > 1$, we break A, B into eight $\frac{n}{2}$ matrices:

$$A_{11} = \begin{pmatrix} 1 & 2 \\ 3 & 8 \end{pmatrix},\ A_{12} = \begin{pmatrix} 1 & 2 \\ 2 & 2 \end{pmatrix},\ A_{21} = \begin{pmatrix} 5 & 1 \\ 6 & 2 \end{pmatrix},\ A_{22} = \begin{pmatrix} 4 & 9 \\ 5 & 0 \end{pmatrix}$$
$$B_{11} = \begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix},\ B_{12} = \begin{pmatrix} 5 & 6 \\ 7 & 8 \end{pmatrix},\ B_{21} = \begin{pmatrix} 9 & 1 \\ 2 & 3 \end{pmatrix},\ B_{22} = \begin{pmatrix} 4 & 5 \\ 6 & 7 \end{pmatrix}$$

Now $\mathrm{MMR}(A_{11}, B_{11}, C_{11}, 2)$ produces eight matrices which are integers. I'll use red indices to distinguish these matrices from the forgoing ones:

$$A_{\color{red}{11}} = 1, A_{\color{red}{12}} = 2, A_{\color{red}{21}} = 3, A_{\color{red}{22}} = 8$$
$$B_{\color{red}{11}} = 1, B_{\color{red}{12}} = 2, B_{\color{red}{21}} = 3, B_{\color{red}{22}} = 4$$

Applying the matrix-multiplication procedure on the integers above, we have

$$C_{\color{red}{11}} = 1 \cdot 1 + 2 \cdot 3 = 7$$
$$C_{\color{red}{12}} = 1 \cdot 2 + 2 \cdot 4 = 10$$
$$C_{\color{red}{21}} = 3 \cdot 1 + 8 \cdot 3 = 24$$
$$C_{\color{red}{22}} = 3 \cdot 2 + 8 \cdot 4 = 38$$

which give us $C_{11} = \mathrm{MMR}(A_{12}, B_{21}, C_{11}, 2) = \begin{pmatrix} 13 & 7 \\ 22 & 8 \end{pmatrix}$.

Applying the matrix multiplication rule again, the sum

$$\begin{aligned} &\mathrm{MMR}(A_{11}, B_{11}, C_{11}, 2) + \mathrm{MMR}(A_{12}, B_{21}, C_{11}, 2) \\ &= \begin{pmatrix} 7 & 10 \\ 24 & 38 \end{pmatrix} + \begin{pmatrix} 13 & 7 \\ 22 & 8 \end{pmatrix} \\ &= \begin{pmatrix} 20 & 17 \\ 46 & 46 \end{pmatrix} \end{aligned}$$

fills in the first quarter of the resultant matrix $A \cdot B = C = \begin{pmatrix} 20 & 17 & x & x \\ 46 & 46 & x & x \\ x & x & x & x \\ x & x & x & x \end{pmatrix}$. We fill in the rest of C in a similar manner.

```
MATRIX-MULTIPLY-RECURSIVE(A, B, C, n)
  if n == 1
  // Base case.
    c_11 = c_11 + a_11 · b_11
    return
  // Divide.
  partition A, B, and C into n/2 × n/2 submatrices A_11,
A_12, A_21, A_22; B_11, B_12, B_21, B_22;
     and C_11, C_12, C_21, C_22; respectively
  // Conquer.
  MATRIX-MULTIPLY-RECURSIVE(A_11, B_11, C_11, n/2)
  MATRIX-MULTIPLY-RECURSIVE(A_11, B_12, C_12, n/2)
  MATRIX-MULTIPLY-RECURSIVE(A_21, B_11, C_21, n/2)
  MATRIX-MULTIPLY-RECURSIVE(A_21, B_12, C_22, n/2)
  MATRIX-MULTIPLY-RECURSIVE(A_12, B_21, C_11, n/2)
  MATRIX-MULTIPLY-RECURSIVE(A_12, B_22, C_12, n/2)
  MATRIX-MULTIPLY-RECURSIVE(A_22, B_21, C_21, n/2)
  MATRIX-MULTIPLY-RECURSIVE(A_22, B_22, C_22, n/2)
```

Block Matrix Multiplication

$$\begin{bmatrix} 152 & 158 & 164 & 170 \\ 504 & 526 & 548 & 570 \\ 856 & 894 & 932 & 970 \\ 1208 & 1262 & 1316 & 1370 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 2 & 3 \\ 4 & 5 & 6 & 7 \\ 8 & 9 & 10 & 11 \\ 12 & 13 & 14 & 15 \end{bmatrix} \times \begin{bmatrix} 16 & 17 & 18 & 19 \\ 20 & 21 & 22 & 23 \\ 24 & 25 & 26 & 27 \\ 28 & 29 & 30 & 31 \end{bmatrix}$$

$$C_{11} = A_{11} \times B_{11} + A_{12} \times B_{21} = \begin{bmatrix} 0 & 1 \\ 4 & 5 \end{bmatrix} \times \begin{bmatrix} 16 & 17 \\ 20 & 21 \end{bmatrix} + \begin{bmatrix} 2 & 3 \\ 6 & 7 \end{bmatrix} \times \begin{bmatrix} 24 & 25 \\ 28 & 29 \end{bmatrix} = \begin{bmatrix} 152 & 158 \\ 504 & 526 \end{bmatrix}$$

Matrix Multiplication: Warmup

To multiply two n-by-n matrices A and B:

- Divide: partition A and B into $\frac{1}{2}n$-by-$\frac{1}{2}n$ blocks.
- Conquer: multiply 8 pairs of $\frac{1}{2}n$-by-$\frac{1}{2}n$ matrices, recursively.
- Combine: add appropriate products using 4 matrix additions.

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

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

$$T(n) = \underbrace{8T(n/2)}_{\text{recursive calls}} + \underbrace{\Theta(n^2)}_{\text{add, form submatrices}} \Rightarrow T(n) = \Theta(n^3)$$

Divide Conquer and Combine Algorithm

Strassen Matrix multiplication: Fast way to multiply 2 matrices

Fast Matrix Multiplication

Key idea. multiply 2-by-2 blocks with only 7 multiplications.

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

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

$$\begin{aligned} C_{11} &= P_5 + P_4 - P_2 + P_6 \\ C_{12} &= P_1 + P_2 \\ C_{21} &= P_3 + P_4 \\ C_{22} &= P_5 + P_1 - P_3 - P_7 \end{aligned}$$

- 7 multiplications.
- $18 = 8 + 10$ additions and subtractions.

Fast Matrix Multiplication

To multiply two n-by-n matrices A and B: [Strassen 1969]

- Divide: partition A and B into $\frac{1}{2}n$-by-$\frac{1}{2}n$ blocks.
- Compute: 14 $\frac{1}{2}n$-by-$\frac{1}{2}n$ matrices via 10 matrix additions.
- Conquer: multiply 7 pairs of $\frac{1}{2}n$-by-$\frac{1}{2}n$ matrices, recursively.
- Combine: 7 products into 4 terms using 8 matrix additions.

Analysis.

- Assume n is a power of 2.
- $T(n)$ = # arithmetic operations.

$$T(n) = \underbrace{7T(n/2)}_{\text{recursive calls}} + \underbrace{\Theta(n^2)}_{\text{add, subtract}} \Rightarrow T(n) = \Theta(n^{\log_2 7}) = O(n^{2.81})$$

Thank You