Задача А3. Быстрее Штрассена!

Вы планируете разработать алгоритм `MULT`, предназначенный для умножения двух квадратных матриц A и B размерности $N \times N$ и асимптотически более эффективный, чем алгоритм Штрассена. Разрабатываемый алгоритм будет также использовать стратегию «разделяй–и–властвуй».

Исходные матрицы A и B разделяются на неизвестное количество фрагментов размера $N/4 \times N/4$ для дальнейшей рекурсивной обработки. Асимптотическая точная граница общих временных затрат на выполнение шагов *CONQUER* и *COMBINE* алгоритма `MULT` — $\Theta(N^2)$. Таким образом, временная сложность алгоритма `MULT` будет описываться рекуррентным соотношением $T(N) = a \cdot T(N/4) + \Theta(N^2)$, где коэффициент a отвечает за количество решаемых подзадач — количество блоков-подматриц размерности $N/4 \times N/4$. Например, для алгоритма Штрассена в соответствии с рекуррентным соотношением $T(N) = 7 \cdot T(N/2) + \Theta(N^2)$ известно, что для каждой задачи решается 7 подзадач *вдвое меньшего* размера.

1. В каком диапазоне должен находиться параметр a разрабатываемого вами алгоритма MULT для того, чтобы в результате он был асимптотически *более эффективным* по временной сложности в сравнении с алгоритмом Штрассена? Обоснуйте свой ответ.

Необходимо проанализировать временные сложности обоих алгоритмов, используя мастер-теорему.

Алгоритм Штрассена

$$T_{\mathrm{S}}(N) = 7 \cdot T_{\mathrm{S}}\left(\frac{N}{2}\right) + \Theta(N^2)$$

Используя мастер-теорему, временная сложность алгоритма Штрассена определяется как:

$$T_{\mathrm{S}}(N) = \Theta\left(N^{\log_2 7}\right)$$

Поскольку $\log_2 7 \approx 2{,}807$, временная сложность составляет примерно $\Theta(N^{2{,}807})$.

Анализ алгоритма MULT

$$T_{\mathrm{M}}(N) = a \cdot T_{\mathrm{M}}\left(\frac{N}{4}\right) + \Theta(N^2)$$

Применим мастер-теорему для общего вида рекуррентного соотношения $T(N) = a \cdot T\left(\frac{N}{b}\right) + f(N)$:

Если $f(N) = O\left(N^{\log_b a - \epsilon}\right)$ для некоторого $\epsilon > 0$, то $T(N) = \Theta\left(N^{\log_b a}\right)$.

Если $f(N) = \Theta\left(N^{\log_b a} \cdot \log^k N\right)$ для некоторого $k \geq 0$, то $T(N) = \Theta\left(N^{\log_b a} \cdot \log^{k+1} N\right)$.

Если $f(N) = \Omega\left(N^{\log_b a+\epsilon}\right)$ для некоторого $\epsilon > 0$ и $a \cdot f\left(\frac{N}{b}\right) \leq c \cdot f(N)$ для некоторого $c < 1$, то $T(N) = \Theta(f(N))$.

Сравнение $f(N)$ и $N^{\log_b a}$
Пусть $s = \log_4 a$. Тогда $N^{\log_4 a} = N^s$.

Случай 1: $s < 2$
$N^s < N^2$
$f(N) = \Theta(N^2) = \Omega\left(N^{s+\epsilon}\right)$ для $\epsilon = 2 - s > 0$
По третьему случаю мастер-теоремы, $T(N) = \Theta(N^2)$
Временная сложность $\Theta(N^2)$, что не лучше алгоритма Штрассена.

Случай 2: $s = 2$
$N^s = N^2$
$f(N) = \Theta(N^{\log_b a})$ - По второму случаю мастер-теоремы, $T(N) = \Theta(N^2 \log N)$
Временная сложность $\Theta(N^2 \log N)$, что лучше $\Theta(N^{2{,}807})$.

Случай 3: $s > 2$
$N^s > N^2$
$f(N) = O\left(N^{s-\epsilon}\right)$ для $\epsilon = s - 2 > 0$
По первому случаю мастер-теоремы, $T(N) = \Theta(N^s)$
Нам нужно, чтобы $s < 2{,}807$, чтобы превзойти алгоритм Штрассена.

Нахождение диапазона a
Требуется $s = \log_4 a < 2{,}807$, то есть:

$$a = 4^s \quad \text{и} \quad s \in (2, 2{,}807)$$

При $s = 2$:

$$a = 4^2 = 16$$

При $s = 2{,}807$:

$$a = 4^{2{,}807} = e^{2{,}807 \cdot \ln 4} = e^{2{,}807 \cdot 2 \ln 2} = e^{5{,}614 \cdot \ln 2} = 2^{5{,}614} \approx 2^5 \cdot 2^{0{,}614} \approx 32 \cdot 1{,}53 \approx 49$$

Вывод

Чтобы обеспечить $T_{\text{M}}(N) = \Theta(N^s)$ асимптотически меньше $\Theta(N^{2{,}807})$:
Параметр a должен удовлетворять неравенству:

$$16 < a < 49$$

Это соответствует диапазону s:

$$2 < s < 2{,}807$$

Для $16 < a < 49$, s находится в диапазоне $2 < s < 2{,}807$, что приводит к временной сложности $\Theta(N^s)$, где $s < 2{,}807$.

Таким образом, $T_{\text{M}}(N) = \Theta(N^s)$ будет асимптотически быстрее, чем $\Theta(N^{2{,}807})$.