Quicksort

Time Complexity:

Average case: $\Theta(n \log n)$

Worst case: $O(n^2)$

Proposition: The expected number of compares to quicksort an array of n distinct elements $a_1 < a_2 < \cdots < a_n$ is $O(n \log n)$.

Proof:

$\mathbf{Pr}[a_i$ and a_j are compared$] = \frac{2}{j-i+1}$

$\mathbb{E}[\#$ of compares$] = \sum_{i=1}^{n} \sum_{j=i+1}^{n} \frac{2}{j-i+1} = 2 \sum_{i=1}^{n} \sum_{j=2}^{n-i+1} \frac{1}{j} \leq 2 \sum_{j=1}^{n} \frac{1}{j} \leq 2n(\ln n + 1)$

In-place sorting

Not a stable one

Application: Matching Nuts and Bolts

Recursion Tree Method

Master Theorem

Divide-and-Conquer

Karatsuba

$T(n) = 3T(\frac{n}{2}) + \Theta(n)$

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

Strassen

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

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

FOCS (Foundation of Computer Science) 2023

Faster Matrix Multiplication via Asymmetric Hashing

Ran Duan *
Tsinghua University

Hongxun Wu †
UC Berkeley

Renfei Zhou ‡
Tsinghua University

April 6, 2023

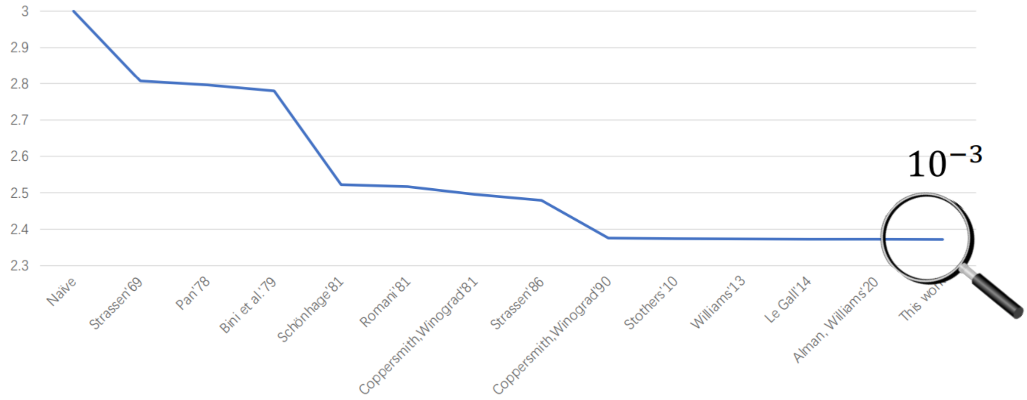
Abstract

Fast matrix multiplication is one of the most fundamental problems in algorithm research. The exponent of the optimal time complexity of matrix multiplication is usually denoted by ω. This paper discusses new ideas for improving the laser method for fast matrix multiplication. We observe that the analysis of higher powers of the Coppersmith-Winograd tensor [Coppersmith & Winograd 1990] incurs a "combination loss", and we partially compensate for it using an asymmetric version of CW's hashing method. By analyzing the eighth power of the CW tensor, we give a new bound of $\omega < 2.371866$, which improves the previous best bound of $\omega < 2.372860$ [Alman & Vassilevska Williams 2020]. Our result breaks the lower bound of 2.3725 in [Ambainis, Filmus & Le Gall 2015] because of the new method for analyzing component (constituent) tensors.

Berkeley Theory Lunch, Feb 2023

Fast Matrix Multiplication

Complexity. $O(n^{\omega})$. $2 \le \omega \le 3$



≡ WIKIPEDIA The Free Encyclopedia | Search Wikipedia | Search | Create account | Log in | •••

Divide-and-conquer algorithm

33 languages

Contents [hide]

Article Talk | Read Edit View history Tools

From Wikipedia, the free encyclopedia

In computer science, **divide and conquer** is an algorithm design paradigm. A divide-and-conquer algorithm recursively breaks down a problem into two or more sub-problems of the same or related type, until these become simple enough to be solved directly. The solutions to the sub-problems are then combined to give a solution to the original problem.

The divide-and-conquer technique is the basis of efficient algorithms for many problems, such as sorting (e.g., quicksort, merge sort), multiplying large numbers (e.g., the Karatsuba algorithm), finding the closest pair of points, syntactic analysis (e.g., top-down parsers), and computing the discrete Fourier transform (FFT).[1]

Designing efficient divide-and-conquer algorithms can be difficult. As in mathematical induction, it is often necessary to generalize the problem to make it amenable to a recursive solution. The correctness of a divide-and-conquer algorithm is usually proved by mathematical induction, and its computational cost is often determined by solving recurrence relations.

≡ WIKIPEDIA The Free Encyclopedia | Search Wikipedia | Search | Create account | Log in | •••

Master theorem (analysis of algorithms)

18 languages

Contents [hide]

Article Talk | Read Edit View history Tools

From Wikipedia, the free encyclopedia

For other theorems called Master theorem, *see Master theorem.*

In the analysis of algorithms, the **master theorem for divide-and-conquer recurrences** provides an asymptotic analysis (using Big O notation) for recurrence relations of types that occur in the analysis of many divide and conquer algorithms. The approach was first presented by Jon Bentley, Dorothea Blostein (née Haken), and James B. Saxe in 1980, where it was described as a "unifying method" for solving such recurrences.[1] The name "master theorem" was popularized by the widely-used algorithms textbook *Introduction to Algorithms* by Cormen, Leiserson, Rivest, and Stein.

Not all recurrence relations can be solved with the use of this theorem; its generalizations include the Akra–Bazzi method.