Gaussian Elimination is not Optimal: Summary

Oliver Vecchini

17th January 2019

Abstract

The following is a summary of the contents, context, and consequences of a seminal paper in theoretical computer science, namely '*Gaussian Elimination is not Optimal*', written by Volker Strassen [1]. It is submitted as a part of the assessed coursework component of the COMP0007 'Directed Reading' module at UCL.

1 Background

Matrices (and the multiplication thereof) are a pervasive aspect of both pure mathematics and computation, underpinning our representation of linear algebra and, by extension, numerous other fields. In a purely mathematical sense, matrix multiplication can be directly related to matrix inversion, (LU) decomposition, determinant calculation, and Gaussian elimination; from a more 'applied' perspective, matrix multiplication is involved in solving systems of linear equations, composition of linear maps, representation of geometric transformations in computer graphics, modelling Markov processes in probability theory, and countless other fields in disciplines with mathematical foundations.

Prior to the 20^{th} century (and as early as the 2^{nd} century BCE, in the form of Gaussian elimination [2]), computation of the matrix product would have been done directly according to its definition; that is, for two square matrices A and B of order n,

$$(AB)_{ij} = \sum_{k=1}^{n} A_{ik} B_{kj}.$$

In this 'naive' algorithm, each element of the matrix is calculated this way, in $O(n)$ elementary operations (as n pairs of elements must be multiplied and the resulting products summed, and where element operations are considered 'elementary'). As there are n^2 elements of the new matrix to be calculated, the overall multiplication takes $O(n^2 n) = O(n^3)$ elementary operations.

It was long known that the best-case upper bound for asymptotic complexity was of the form $O(n^{2+c})$ for some $c >= 0$, i.e. that the exponent of n would have to be at least 2 since all n^2 elements of each matrix were being considered. It was long assumed (and even 'proven' [3] for Gaussian elimination, which is equivalent to matrix multiplication as stated above) that the bound was $O(n^3)$.

Although some improvement arrived (e.g. from S. Winograd in 1967 [4], providing a new algorithm with more elementary addition and less elementary multiplication, and thus a reasonable speed-up in systems where the former was faster than the latter (up to 30% in contemporary systems [5])), asymptotic complexity remained the same at $O(n^3)$. Meaningful improvement of computation speed (and thus throughput of all the applications listed above) would require improvement of the asymptotic bound for execution speed.

2 Contents of the article

The article not only refutes the claim of ordinary Gaussian elimination being optimal (by showing that the naive implementation of matrix multiplication is suboptimal, combined with being equivalent to Gaussian elimination), but shows this by providing an algorithm of complexity bound $O(n^{log_2 7})$ $(= O(n^{2.81\cdots}))$.

Strassen's algorithm uses Winograd's idea [4] of embedding (i.e. padding with rows/columns of zeros) matrices A, B, and AB into matrices of order 2^k for some $k \in \mathbb{N}$, and partitioning each of these into four submatrices each of order 2^{k-1}, so that the multiplication process involves multiplying two matrices of order 2, where the elements of each matrix are the aforementioned submatrices. As such, using the identity $C = AB$,

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

The algorithm then uses various identities (arising from matrix symmetries [6]) to calculate the product using only seven submatrix multiplications, compared to eight as in Winograd's algorithm. The article explains that, with a recursive depth of k (i.e. k 'levels' of recursive calls to the algorithm, to multiply the submatrices with eachother), the algorithm subsequently executes $O(7^k)$ elementary multiplications and additions, and that substituting back, and discounting multiplying zero rows/ columns, yields an $O(n^{\log_2 7})$ complexity bound. An equivalent algorithm is provided for the case of matrix inversion.

With respect to proof, Strassen presents the framework for proving the claim of the complexity bound in each case of multiplication and inversion algorithm, but leaves the induction proof demonstrating the crucial bounds to the reader; the proofs are still largely complete as presented, since the induction segments left to the reader are relatively manageable instances of weak induction over a single variable.

Strassen finishes by noting that this concept can also be applied to solving systems of linear equations or computation of determinant; while these are not explicitly proven, it is again reasonably persuasive, given the equivalence between these and matrix multiplication/inversion.

3 Legacy

The significant impacts of the publication of Strassen's paper were twofold. The first was the provision of a new algorithm for matrix multiplication, potentially increasing the throughput of countless scientific and industrial calculations across a range of fields. However, in reality, the constants associated with the complexity formula (obscured by big-O notation) meant that the algorithm was only suited for matrices of very large order (as evidenced by benchmarks of the time, with Strassen's algorithm only optimal for matrix orders in the hundreds on contemporary [5] and modern [7] systems). In addition, its relative advantage may have declined, due to naive multiplication being heavily optimised at the hardware level, in particular for graphics applications requiring large quantities of geometric transformations per second.

The second major impact of the article was establishing that the previously trusted bound of $O(n^3)$ execution time was discarded, invigorating further research into possible improvements. Among the most significant advancements was the Coppersmith-Winograd algorithm published in 1990 [8], providing a bound of $O(n^{2.376})$. However, the constants of the complexity formula hidden by big-O notation are so large that the order of matrices to be multiplied would be enormous enough to be irrelevant to modern applications [9]. Despite this, further algorithms in this topic are being actively researched, and the principles of Strassen's algorithm are used as the foundations for several of them [9].

4 In Review

For a topic requiring little fundamental knowledge aside from basic linear algebra, it is unfortunate that Strassen's article should remain somewhat obtuse in its explanation and in its relation to the title presented (failing to explicitly note the congruence between Gaussian elimination and the algorithms presented), seemingly chosen solely to catch the reader's attention. The inability to fully write out non-trivial proofs is irritating, but possibly an artefact of a different style of paper that was ubiquitous 50 years ago compared to today, as it does seem consistent with other papers of that age by renowned authors [10]. This would also be understandable if the goal was to maximise brevity, and keep the paper as simple as possible. In any case, minor quibbles aside, the revolutionary claims of the paper and its impact on a ubiquitous calculational task make it understandable that the paper would be considered a classic.

References

[1] Volker Strassen. Gaussian elimination is not optimal. *Numerische mathematik*, 13(4):354–356, 1969.

[2] Ronald Calinger, Joseph E Brown, and Thomas R West. *A contextual history of mathematics: To Euler.* Prentice Hall Upper Saddle River, NJ, 1999.

[3] VV Klyuev and NI Kokovkin-Shcherbak. On the minimization of the number of arithmetic operations for the solution of linear algebraic systems of equations. Technical report, STANFORD UNIV CALIF DEPT OF COMPUTER SCIENCE, 1965.

[4] Shmuel Winograd. A new algorithm for inner product. *IEEE Transactions on Computers*, 100(7):693–694, 1968.

[5] Richard P Brent. Algorithms for matrix multiplication. Technical report, STANFORD UNIV CA DEPT OF COMPUTER SCIENCE, 1970.

[6] Ann Q. Gates and Vladik Kreinovich. Strassen's algorithm made (somewhat) more natural: A pedagogical remark. *Bulletin of the EATCS*, 73:142–145, 2001.

[7] M Juby. R. vijaya k.,"comparative study of strassen's matrix multiplication algorithm". *International Journal of Computer Science And Technology IJCST*, 3(1), 2012.

[8] Don Coppersmith and Shmuel Winograd. Matrix multiplication via arithmetic progressions. *Journal of symbolic computation*, 9(3):251–280, 1990.

[9] Sara Robinson. Toward an optimal algorithm for matrix multiplication. *SIAM news*, 38(9):1–3, 2005.

[10] Donald E Knuth. On the translation of languages from left to right. *Information and control*, 8(6):607–639, 1965.