Checking Matrix Identities (2006; Buhrman, Špalek)

Ashwin Nayak, University of Waterloo and Perimeter Institute for Theoretical Physics,
www.math.uwaterloo.ca/~anayak

1 Synonyms

Matrix product verification.

2 Index terms

Fingerprinting, linear algebra, matrix multiplication, program testing, quantum algorithm, quantum walk.

3 Problem definition

Let A, B, C be three given matrices of dimension $n \times n$ over a field, where C is claimed to be the matrix product AB. The straightforward method of checking whether $C = AB$ is to multiply the matrices A, B, and compare the entries of the result with those of C. This takes time $O(n^\omega)$, where ω is the "exponent of matrix multiplication". It is evident from the definition of the matrix multiplication operation that $2 \leq \omega \leq 3$. The best known bound on ω is 2.376 [4].

Here, and in the sequel, "time" is taken to mean "number of arithmetic operations" over the field (or other algebraic structure to which the entries of the matrix belong). Similarly, in stating space complexity, the multiplicative factor corresponding to the space required to represent elements of the algebraic structure is suppressed.

Surprisingly, matrix multiplication can be circumvented by using a randomized "fingerprinting" technique due to Freivalds [5], and the matrix product can be checked in time $O(n^2)$ with one-sided bounded probability of error. This algorithm extends, in fact, to matrices over any *integral domain* [3] and the number of random bits used may be reduced to $\log \frac{n}{\epsilon} + O(1)$ for an algorithm that makes one-sided probabilistic error at most ϵ [8]. (All logarithms in this article are taken to base 2.) The fingerprinting technique has found numerous other applications in theoretical computer science (see, for example, Ref. [10]).

Buhrman and Špalek consider the complexity of checking matrix products on a quantum computer.

Problem 1 *(Matrix product verification)*
INPUT: *Matrices A, B, C of dimension $n \times n$ over an integral domain.*
OUTPUT: EQUAL *if $C = AB$, and* NOT EQUAL *otherwise.*

They also study the verification problem over the Boolean algebra $\{0,1\}$ with operations $\{\vee,\wedge\}$, where the fingerprinting technique does not apply.

As an application of their verification algorithms, they consider multiplication of sparse matrices.

Problem 2 *(Matrix multiplication)*
INPUT: *Matrices A, B of dimension $n \times n$ over an integral domain or the Boolean algebra $\{0,1\}$.*
OUTPUT: *The matrix product $C = AB$ over the integral domain or the Boolean algebra.*

4 Key results

Ambainis, Buhrman, Høyer, Karpinski, and Kurur [2] first studied matrix product verification in the quantum mechanical setting. Using a recursive application of the Grover search algorithm [6], they gave an $O(n^{7/4})$ algorithm for the problem. Buhrman and Špalek improve this runtime by adapting search algorithms based on quantum walk that were recently discovered by Ambainis [1] and Szegedy [11].

Let $W = \{(i,j)|(AB-C)_{i,j} \neq 0\}$ be the set of coordinates where C disagrees with the product AB, and let W' be the largest independent subset of W. (A set of coordinates is said to be *independent* if no row or column occurs more than once in the set.) Define $q(W) = \max\{|W'|, \min\{|W|, \sqrt{n}\}\}$.

Theorem 4.1 *Consider Problem 1. There is a quantum algorithm that always returns* EQUAL *if $C = AB$, returns* NOT EQUAL *with probability at least $\frac{2}{3}$ if $C \neq AB$, and has worst case run-time $O(n^{5/3})$, expected run-time $O(n^{2/3}/q(W)^{1/3})$, and space complexity $O(n^{5/3})$.*

Buhrman and Špalek state their results in terms of "black-box" complexity or "query complexity", where the entries of the input matrices A, B, C are provided by an oracle. The measure of complexity here is the number of oracle calls (queries) made. The query complexity of their quantum algorithm is the same as the run time in the above theorem. They also derive a lower bound on the query complexity of the problem.

Theorem 4.2 *Any bounded-error quantum algorithm for Problem 1 has query complexity $\Omega(n^{3/2})$.*

When the matrices A, B, C are Boolean, and the product is defined over the operations $\{\vee,\wedge\}$, an algorithm with run-time/query complexity $O(n^{3/2})$ may be derived from an algorithm for AND-OR trees [7]. This has space complexity $O((\log n)^3)$.

All the quantum algorithms may be generalized to handle rectangular matrix product verification, with appropriate modification to the run-time and space complexity.

5 Applications

Using binary search along with the algorithms in the previous section, the position of a wrong entry in a matrix C (purported to be the product AB) can be located, and then corrected. Buhrman and Špalek use this in an iterative fashion to obtain a matrix multiplication algorithm, starting from the guess $C = 0$. When the product AB is a sparse matrix, this leads to a quantum matrix multiplication scheme that is, for some parameters, faster than known classical schemes.

Theorem 5.1 *For any $n \times n$ matrices A, B over an integral domain, the matrix product $C = AB$ can be computed by a quantum algorithm with polynomially small error probability in expected time*

$$\mathrm{O}(1) \cdot \begin{cases} n \log n \cdot n^{2/3} w^{2/3} & \text{when } 1 \leq w \leq \sqrt{n}, \\ n \log n \cdot \sqrt{n}\, w & \text{when } \sqrt{n} \leq w \leq n, \text{ and} \\ n \log n \cdot n\sqrt{w} & \text{when } n \leq w \leq n^2, \end{cases}$$

where w is the number of non-zero entries in C.

A detailed comparison of this quantum algorithm with classical ones may be found in Ref. [3].

A subsequent quantum walk based algorithm due to Magniez, Nayak, Roland, and Santha [9] finds a wrong entry in the same run-time as in Theorem 4.1, without the need for binary search. This improves the run-time of the quantum algorithm for matrix multiplication described above slightly.

Since Boolean matrix products can be verified faster, boolean matrix products can be computed in expected time $\mathrm{O}(n^{3/2}\sqrt{w})$, where w is the number of '1' entries in the product.

All matrix product algorithms presented here may be used for multiplication of rectangular matrices as well, with appropriate modifications.

6 Cross references

Amplitude Amplification (00008), Element Distinctness (00015), Quantization of Markov Chains (00016).

7 Recommended reading

[1] Andris Ambainis. Quantum walk algorithm for Element Distinctness. In *Proceedings of the 45th Symposium on Foundations of Computer Science*, pages 22–31, 2004.

[2] Andris Ambainis, Harry Buhrman, Peter Høyer, Marek Karpinski, and P. Kurur. Quantum matrix verification. Unpublished manuscript, 2002.

[3] Harry Buhrman and Robert Špalek. Quantum verification of matrix products. In *Proceedings of 17th ACM-SIAM Symposium on Discrete Algorithms*, pages 880–889, 2006.

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

[5] Rūsiņš Freivalds. Fast probabilistic algorithms. In *Proceedings of the 8th Symposium on Mathematical Foundations of Computer Science*, pages 57–69, 1979.

[6] Lov K. Grover. A fast quantum mechanical algorithm for database search. In *Proceedings of the 28th ACM Symposium on the Theory of Computing*, pages 212–219, 1996.

[7] Peter Høyer, Michele Mosca, and Ronald de Wolf. Quantum search on bounded-error inputs. In *Proceedings of the 30th International Colloquium on Automata, Languages and Programming*, volume 2719 of *Lecture Notes in Computer Science*, pages 291–299, 2003.

[8] Tracy Kimbrel and Rakesh Kumar Sinha. A probabilistic algorithm for verifying matrix products using $O(n^2)$ time and $\log_2 n + O(1)$ random bits. *Information Processing Letters*, 45(2):107–110, 1993.

[9] Frédéric Magniez, Ashwin Nayak, Jérémie Roland, and Miklos Santha. Search via quantum walk. Technical Report quant-ph/0608026, ArXiv.org Preprint Archive, `http://www.arxiv.org/abs/quant-ph/`, August 2006. To appear in STOC 2007.

[10] Rajeev Motwani and Prabhakar Raghavan. *Randomized Algorithms*. Cambridge University Press, 1995.

[11] Mario Szegedy. Quantum speed-up of Markov chain based algorithms. In *Proceedings of the 45th IEEE Symposium on Foundations of Computer Science*, pages 32–41, 2004.