Improved Output-Sensitive Quantum Algorithms for Boolean Matrix Multiplication

François Le Gall
Department of Computer Science
Graduate School of Information Science and Technology
The University of Tokyo
legall@is.s.u-tokyo.ac.jp

Abstract

We present new quantum algorithms for Boolean Matrix Multiplication in both the time complexity and the query complexity settings. As far as time complexity is concerned, our results show that the product of two $n \times n$ Boolean matrices can be computed on a quantum computer in time $\tilde{O}(n^{3/2} + n\ell^{3/4})$, where ℓ is the number of non-zero entries in the product, improving over the output-sensitive quantum algorithm by Buhrman and Špalek (SODA'06) that runs in $\tilde{O}(n^{3/2}\sqrt{\ell})$ time. This is done by constructing a quantum version of a recent classical algorithm by Lingas (ESA'09), using quantum techniques such as quantum counting to exploit the sparsity of the output matrix. As far as query complexity is concerned, our results improve over the quantum algorithm by Vassilevska Williams and Williams (FOCS'10) based on a reduction to the triangle finding problem. One of the main contributions leading to this improvement is the construction of a quantum algorithm for triangle finding tailored especially for the tripartite graphs appearing in the reduction.

Boolean matrix multiplication, where addition is interpreted as a logical OR and multiplication as a logical AND, is a fundamental problem in computer science. Algorithms for Boolean matrix multiplication have found applications in many areas and are, for example, used to construct efficient algorithms for computing the transitive closure of a graph [7, 8, 15], recognizing context-free languages [16, 20], detecting if a graph contains a triangle [11], solving all-pairs path problems [6, 9, 18, 19], or speeding up data mining tasks [1].

The product of two Boolean $n \times n$ matrices A and B can be trivially computed in time $O(n^3)$. The best known classical algorithm is obtained by seeing the matrices A and B as integer matrices, computing the integer matrix product, and converting the product matrix to a Boolean matrix. Using the algorithm by Coppersmith and Winograd [5] for integer matrix multiplication, this gives an algorithm for Boolean matrix multiplication with time complexity $O(n^{2.376})$. This approach has nevertheless several disadvantages, the main one being that Coppersmith-Winograd's algorithm can be hard to implement in practice. Partly for this reason, much effort has focused on understanding whether Boolean matrix multiplication can be done in $o(n^3)$ time by combinatorial algorithms, i.e., classical algorithms that do not rely on a product of matrices over rings. A maybe more fundamental reason for investigating this question is that a fast combinatorial algorithm for matrix multiplication over the semi-ring (OR, AND) would possibly generalize to other semi-rings, and especially to semi-rings such as $(\min, +)$ related to a multitude of problems over weighted graphs such as the all-pairs shortest paths problem, over which no subcubic time multiplication algorithm is available. Unfortunately, there have been little progress on this question. The best known combinatorial algorithm has time complexity $O(n^3/\log^{2.25}(n))$ and has been discovered recently by Bansal and Williams [3], improving the "four Russians" algorithm [2] proposed decades ago.

In the quantum computation model, there exist subcubic-time algorithms for Boolean matrix multiplication that do not rely on integer matrix multiplication. Indeed, the product of two $n \times n$ Boolean matrices A and B can be easily computed in time $\tilde{O}(n^{2.5})$: for each pair of indexes $i, j \in \{1, 2, \ldots, n\}$, check if there exists an index $k \in \{1, \ldots, n\}$ such that $A[i,k] = B[k,j] = 1$ in time $\tilde{O}(\sqrt{n})$ using Grover's quantum search algorithm [10]. Buhrman and Špalek [4] observed that a similar argument can be used

to design an efficient output-sensitive quantum algorithm for Boolean matrix multiplication. The idea is to perform a quantum search over the couples (i, j) on top of the Grover search for k. This leads to a quantum algorithm that computes the product AB in $\tilde{O}(n^{3/2}\sqrt{\ell})$ time, where ℓ denotes the number of non-zero entries in AB. Recently, Vassilevska Williams and Williams [21] constructed faster output-sensitive quantum algorithms for Boolean matrix multiplications in the query complexity setting (where the complexity under consideration is the number of queries to the entries of the input matrices A and B). Using a query-efficient triangle finding quantum algorithm by Magniez, Santha and Szegedy [14] and ideas from Lingas [13], they obtained an algorithm with query complexity $\tilde{O}(\min(n^{1.3}\ell^{17/30}, n^2 + n^{13/15}\ell^{47/60}))$.

Statement of our results and comparison with previous work. In this work we present new quantum algorithms for Boolean Matrix Multiplication in both the time complexity and the query complexity settings. Our first result is stated in the following theorem.

Theorem 1. *There exists a quantum algorithm that computes the product of two $n \times n$ Boolean matrices with time complexity $\tilde{O}(n^{3/2})$ if $1 \leq \ell \leq n^{2/3}$ and $\tilde{O}(n\ell^{3/4})$ if $n^{2/3} \leq \ell \leq n^2$, where ℓ denotes the number of non-zero entries in the product.*

This new algorithm improves the quantum algorithm by Buhrman and Špalek [4] for any value of ℓ other than $\ell = O(\text{poly}(\log n))$ or $\ell = \Theta(n^2)$. For example, for $\ell = n^{1.2}$, the complexity of our algorithm is $\tilde{O}(n^{1.9})$, while the complexity of the latter algorithm is $\tilde{O}(n^{2.1})$. For $\ell = O(\text{poly}(\log n))$ or $\ell = \Theta(n^2)$ our upper bounds are similar to the bounds achieved by Buhrman and Špalek's algorithm.

Our quantum algorithm is always faster than the standard classical output-sensitive combinatorial algorithm for Boolean matrix multiplication, which has time complexity $\tilde{O}(n\ell + n^2)$ and uses a column-row approach (see [17] and the discussion in [13]). The best known classical algorithms for output-sensitive Boolean matrix multiplication are based on Coppersmith-Winograd's algorithm: Amossen and Pagh [1] constructed an algorithm with time complexity $O(n^{1.72}\ell^{0.41} + n^{4/3}\ell^{2/3})$, while Lingas [13] constructed an algorithm with time complexity $\tilde{O}(n^2\ell^{0.188})$. The quantum algorithm of Theorem 1 beats both of them for any value $\ell \leq n^{1.779}$.

Our second result deals with the query complexity of Boolean matrix multiplication.

Theorem 2. *There exists a quantum algorithm that computes the product of two $n \times n$ Boolean matrices with query complexity*

$$\begin{cases} \tilde{O}(n^{1.3}\sqrt{\ell}) & \text{if } 1 \leq \ell \leq n^{1/5} \\ \tilde{O}(n^{9/7}\ell^{4/7}) & \text{if } n^{1/5} \leq \ell \leq n^{3/8} \\ \tilde{O}\left(n^{3/2} + \min(n^{13/14}\ell^{3/4}, n^{3/2}\ell^{1/4}) + n^{6/7}\ell^{11/14}\right) & \text{if } n^{3/8} \leq \ell \leq n^2 \end{cases},$$

where ℓ denotes the number of non-zero entries in the product.

The bounds of Theorem 2 are illustrated in Figure 1. Our new quantum algorithm improves the quantum algorithm by Vassilevska Williams and Williams [21] for any value of ℓ other than $\ell = O(\text{poly}(\log n))$ (if $\ell = O(\text{poly}(\log n))$ then our bounds are the same). For instance, in the case $\ell = n^{1.2}$, the algorithm of Theorem 2 uses $\tilde{O}(n^{1.8})$ queries, while the quantum algorithm from [21] uses $\tilde{O}(n^{1.98})$ queries. Let us mention that, if $\ell = \Theta(n^2)$, then we obtain query complexity $\tilde{O}(n^{2+3/7})$, slightly improving the bound $\tilde{O}(n^{2+13/30})$ from [21]. One may argue that query complexity upper bounds such as the later two bounds are not meaningful since $2n^2$ queries are enough to obtain all the entries of the two input matrices. Such a strategy would however a priori require storing all the $2n^2$ entries, while ours actually uses a total amount of $\tilde{O}(n^{9/14})$ bits and qubits of memory (slightly improving the $\tilde{O}(n^{2/3})$ space complexity of the algorithm in [21]). This situation can also be put in perspective by considering the trade-off $Q^2S = \Omega(n^5)$ proved in [12] between the query complexity Q and the space complexity S of any quantum algorithm computing the Boolean matrix product of two $n \times n$ matrices.

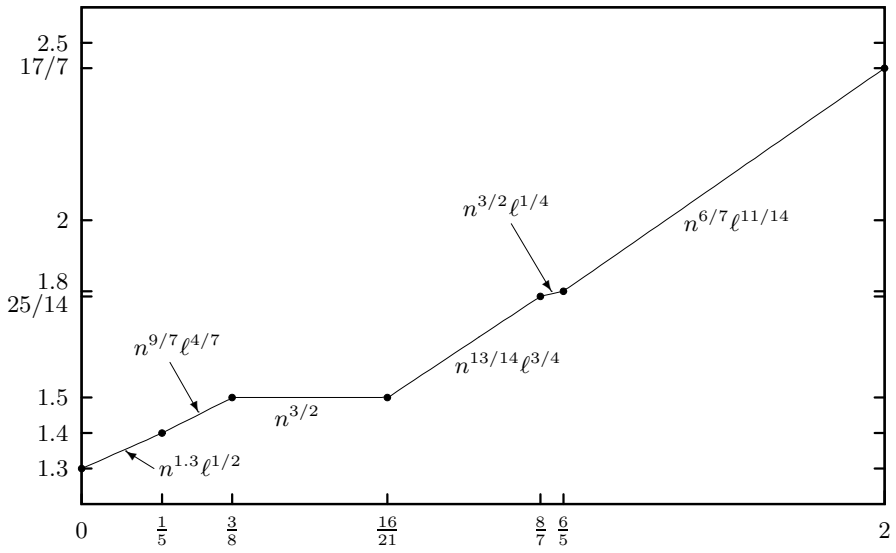

Figure 1: The upper bounds on the query complexity of matrix multiplication given in Theorem 2. The horizontal axis represents the logarithm of ℓ with respect to basis n (i.e., the value $\log_n(\ell)$). The vertical axis represents the logarithm of the upper bounds of Theorem 2 with respect to basis n.

Overview of our techniques. As far as our query complexity results are concerned, our starting point is the reduction by Vassilevska Williams and Williams [21]. Let us first consider the case where the product $C = AB$ of the two $n \times n$ Boolean matrices A and B is dense (i.e., $\ell \approx n^2$). The reduction of [21] can be informally described as reducing the computation of C to about n^2 instances of the triangle finding problem, each instance being over a graph of size $n^{1/3}$ (more precisely, over a tripartite graph with vertex sets (I', J', K') such that $|I'| = |J'| = |K'| = n^{1/3}$). By using the quantum algorithm by Magniez, Santha and Szegedy [14] that finds with $\tilde{O}(n^{1.3})$ queries a triangle in a graph of size n, this gives a quantum algorithm for matrix multiplication with query complexity $\tilde{O}(n^{2+13/30})$. Our first idea is that this reduction can be improved by considering tripartite graphs over three sets of unbalanced size. In order to take advantage of this idea, we then design a version of the triangle algorithm in [14] tailored especially for such tripartite graphs. By optimizing the parameters, we obtain a reduction from the computation of C to n^2 instances of the triangle finding problem, each instance being over a tripartite graph with vertex sets (I', J', K') such that $|I'| = |J'| = n^{9/28}$ and $|K'| = n^{5/14}$, with an overall cost of $\tilde{O}(n^{2+3/7})$ queries. With some more work we derive a similar algorithm for the output-sensitive case, with complexity $\tilde{O}(n^{1.3}\sqrt{\ell})$ if $1 \leq \ell \leq n^{1/5}$, and $\tilde{O}(n^{9/7}\ell^{4/7})$ if $n^{1/5} \leq \ell \leq n^2$.

In the classical setting, Lingas [13] has shown how to reduce the output-sensitive computation of a product of two $n \times n$ matrices to the output-sensitive computation of products of smaller matrices, at the price of a $\tilde{O}(n^2)$-time additive cost representing preprocessing and postprocessing steps. This strategy was also used in [21], with the same preprocessing/postprocessing cost, to obtain the $\tilde{O}(n^2 + n^{13/45}\ell^{47/60})$-query quantum algorithm mentioned above. We obtain Theorems 1 and 2 by constructing a quantum version of Lingas' reduction where both the preprocessing cost and the postprocessing cost are reduced. More precisely, we show that, by using quantum search and quantum counting techniques, $\tilde{O}(n^{3/2})$ time is enough in the quantum setting to perform most of the preprocessing/postprocessing operations used in [13], and that the remaining operations can be treated as subroutines in matrix multiplication quantum algorithms without generating a significant additional cost. Finally, the products of the resulting smaller matrices are computed using a simple time-efficient quantum algorithm inspired by the work of Buhrman and Špalek [4] to obtain Theorem 1, and using the query-efficient quantum algorithms described in the previous paragraph to obtain Theorem 2.

References

[1] AMOSSEN, R. R., AND PAGH, R. Faster join-projects and sparse matrix multiplications. In *Proceedings of Database Theory - ICDT* (2009), pp. 121–126.

[2] ARLAZAROV, V. L., DINIC, E. A., KRONROD, M. A., AND FARADZEV, I. A. On economical construction of the transitive closure of a directed graph. *Soviet Mathematics Doklady (English translation) 11*, 5 (1970), 1209–1210.

[3] BANSAL, N., AND WILLIAMS, R. Regularity lemmas and combinatorial algorithms. In *Proceedings of the 50th Annual Symposium on Foundations of Computer Science* (2009), pp. 745–754.

[4] BUHRMAN, H., AND SPALEK, R. Quantum verification of matrix products. In *Proceedings of the 17th Annual Symposium on Discrete Algorithms* (2006), pp. 880–889.

[5] COPPERSMITH, D., AND WINOGRAD, S. Matrix multiplication via arithmetic progressions. *Journal of Symbolic Computation 9*, 3 (1990), 251–280.

[6] DOR, D., HALPERIN, S., AND ZWICK, U. All-pairs almost shortest paths. *SIAM Journal on Computing 29*, 5 (2000), 1740–1759.

[7] FISCHER, M. J., AND MEYER, A. R. Boolean matrix multiplication and transitive closure. In *Proceedings of the 12th Annual Symposium on Switching and Automata Theory* (1971), pp. 129–131.

[8] FURMAN, M. E. Application of a method of fast multiplication of matrices in the problem of finding the transitive closure of a graph. *Soviet Mathematics Doklady (English translation) 11*, 5 (1970), 1252.

[9] GALIL, Z., AND MARGALIT, O. All pairs shortest distances for graphs with small integer length edges. *Information and Computation 134*, 2 (1997), 103–139.

[10] GROVER, L. K. A fast quantum mechanical algorithm for database search. In *Proceedings of the 28th Annual Symposium on the Theory of Computing* (1996), pp. 212–219.

[11] ITAI, A., AND RODEH, M. Finding a minimum circuit in a graph. *SIAM Journal on Computing 7*, 4 (1978), 413–423.

[12] KLAUCK, H., SPALEK, R., AND DE WOLF, R. Quantum and classical strong direct product theorems and optimal time-space tradeoffs. *SIAM Journal on Computing 36*, 5 (2007), 1472–1493.

[13] LINGAS, A. A fast output-sensitive algorithm for boolean matrix multiplication. *Algorithmica* (2011), 36–50.

[14] MAGNIEZ, F., SANTHA, M., AND SZEGEDY, M. Quantum algorithms for the triangle problem. *SIAM Journal on Computing 37*, 2 (2007), 413–424.

[15] MUNRO, J. I. Efficient determination of the transitive closure of a directed graph. *Information Processing Letters 1*, 2 (1971), 56–58.

[16] RYTTER, W. Fast recognition of pushdown automaton and context-free languages. *Information and Control 67*, 1-3 (1985), 12–22.

[17] SCHNORR, C.-P., AND SUBRAMANIAN, C. R. Almost optimal (on the average) combinatorial algorithms for boolean matrix product witnesses, computing the diameter (extended abstract). In *Proceedings of the 2nd workshop on Randomization and Approximation Techniques in Computer Science (RANDOM)* (1998), pp. 218–231.

[18] SEIDEL, R. On the all-pairs-shortest-path problem in unweighted undirected graphs. *Journal of Computer and System Sciences 51*, 3 (1995), 400–403.

[19] SHOSHAN, A., AND ZWICK, U. All pairs shortest paths in undirected graphs with integer weights. In *Proceedings of the 40th Annual Symposium on Foundations of Computer Science* (1999), pp. 605–615.

[20] VALIANT, L. G. General context-free recognition in less than cubic time. *Journal of Computer and System Sciences 10*, 2 (1975), 308–315.

[21] VASSILEVSKA WILLIAMS, V., AND WILLIAMS, R. Subcubic equivalences between path, matrix and triangle problems. In *Prooceedings of the 51th Annual Symposium on Foundations of Computer Science* (2010), pp. 645–654.