Quantum Algorithms for Finding Constant-sized Sub-hypergraphs*

François Le Gall* Harumichi Nishimura† Seiichiro Tani‡

*Graduate School of Information Science and Technology, The University of Tokyo
†Graduate School of Information Science, Nagoya University
‡NTT Communication Science Laboratories, NTT Corporation

Quantum query complexity is a model of quantum computation, in which the cost of computing a function is measured by the number of queries that are made to the input given as a black-box. In this model, it was exhibited in the early stage of quantum computing research that there exist quantum algorithms superior to the classical counterparts, such as Deutsch and Jozsa's algorithm, Simon and Shor's period finding algorithms, and Grover's search algorithm. Extensive studies following them have invented a lot of powerful upper bound (i.e., algorithmic) techniques such as variations/generalizations of Grover's search algorithm or quantum walks. Although these techniques give tight bounds for many problems, there are still quite a few cases for which no tight bounds are known. Intensively studied problems among them are the k-distinctness problem [1, 3, 4] and the triangle finding problem [2, 6, 8, 12, 14, 10].

A recent breakthrough is the concept of learning graph introduced by Belovs [2]. This concept enables one to easily find a special form of feasible solutions to the minimization form (i.e., the dual form) of the general adversary bound [7, 15], and makes possible to detour the need of solving a semidefinite program of exponential size to find a non-trivial upper bound. Indeed, Belovs [2] improved the long-standing $\tilde{O}(n^{13/10})$ upper bound [14] (which was slightly improved to $O(n^{13/10})$ [13]) of the triangle finding problem to $O(n^{35/27})$. His idea was generalized by Lee, Magniez and Santha [11] and Zhu [23] to obtain a quantum algorithm that finds a constant-sized subgraph with complexity $o(n^{2-2/k})$, improving the previous best bound $O(n^{2-2/k})$ [14], where k is the size of the subgraph. Subsequently, Lee, Magniez and Santha [12] constructed a triangle finding algorithm with quantum query complexity $O(n^{9/7})$. This bound was later shown by Belovs and Rosmanis [5] to be the best possible bound attained by the family of quantum algorithms whose complexities depend only on the index set of 1-certificates (very recently, Le Gall [10] broke this $n^{9/7}$-barrier via combinatorial arguments to obtain the current best quantum upper bound of $\tilde{O}(n^{5/4})$). Ref. [12] also gave a framework of quantum algorithms for finding a constant-sized subgraph, based on which they showed that associativity testing (testing if a binary operator over a domain of size n is associative) has quantum query complexity $O(n^{10/7})$.

Jeffery, Kothari and Magniez [8] cast the idea of the above triangle finding algorithms into the framework of quantum walks (called nested quantum walks) by recursively performing the quantum walk algorithm given by Magniez, Nayak, Roland and Santha [13] (which extended two seminal works for quantum walk algorithms by Szegedy [18] and Ambainis [1]). Indeed, they presented two quantum-walk-based triangle finding algorithms of complexities $\tilde{O}(n^{35/27})$ and $\tilde{O}(n^{9/7})$, respectively. The nested quantum walk framework was further employed in [4] (but in a different way from [8]) to obtain $\tilde{O}(n^{5/7})$ complexity for the 3-distinctness problem. This achieves the best known upper bound (up to poly-logarithmic factors), which was first obtained with the learning-graph-based approach [3].

The triangle finding problem also plays a central role in several areas beside query complexity, and it has been recently discovered that faster algorithms for (weighted versions of) triangle finding would lead to faster algorithms for matrix multiplication [9, 20], the 3SUM problem [19], and for Max-2SAT [21, 22]. In particular, Max-2SAT over n variables is reducible to finding a triangle with maximum weight over $O(2^{n/3})$ vertices; in this context, although the final goal is a *time-efficient* classical or quantum algorithm that finds a triangle with maximum weight, studying triangle finding in the query complexity model is a first step toward this goal.

*This paper was also presented at 20th International Computing and Combinatorics Conference (COCOON 2014) [LNCS vol. 8591. pp. 429–440]. A full version is available at arXiv:1310.4127v2 [quant-ph].

Along this line of research, this paper studies the problem of finding a 4-clique (i.e., the complete 3-uniform hypergraph with 4 vertices) in a 3-uniform hypergraph, a natural generalization of finding a triangle in an ordinary graph (i.e., a 2-uniform hypergraph). Our initial motivation comes from the complexity-theoretic importance of the problem. Indeed, while it is now well-known that Max-3SAT over n variables is reducible to finding a 4-clique with maximum weight in a 3-uniform hypergraph of $O(2^{n/4})$ vertices, no efficient classical algorithm for 4-clique finding has been discovered so far. Constructing query-efficient algorithms for this problem can be seen as a first step to investigate the possibility of faster (in the time complexity setting) classical or quantum algorithms for Max-3SAT.

Concretely, and more generally, this paper gives a framework based on quantum walks for finding any constant-sized sub-hypergraph in a 3-uniform hypergraph. This is an extension of the learning-graph-based algorithm in [12] to the hypergraph case in terms of a nested quantum walk [8]. We illustrate this methodology by constructing a quantum algorithm that finds a 4-clique in a 3-uniform hypergraph with query complexity $\tilde{O}(n^{241/128}) = O(n^{1.883})$, while naïve Grover search over the $\binom{n}{4}$ combinations of vertices only gives $O(n^2)$. As another application, we also construct a quantum algorithm that determines if a ternary operator is associative using $\tilde{O}(n^{169/80}) = O(n^{2.113})$ queries, while naïve Grover search needs $O(n^{2.5})$ queries.

In the course of designing the quantum walk framework, we introduce several new technical ideas (outlined below) for analyzing nested quantum walks to cope with difficulties that do not arise in the 2-uniform case (i.e., ordinary graphs), such as the fact that the size of the random subset taken in an inner walk may vary depending on the random subsets taken in outer walks. We believe that these ideas may be applicable to various problems beyond sub-hypergraph finding.

Statement of our results. We state the main result (the algorithm which finds a constant-size sub-hypergraph in a hypergraph) in terms of loading schedules, which generalize the concept of loading schedules for graphs introduced, in the learning graph framework, by Lee, Magniez and Santha [12], and used in the framework of nested quantum walks by Jeffery, Kothari and Magniez [8].

Let H be a 3-uniform hypergraph with κ vertices. We identify the set of vertices of H with the set $\Sigma_1 = \{1,\ldots,\kappa\}$. We identify the set of (unordered) pairs of vertices of H with the set $\Sigma_2 = \{\{1,2\},\{1,3\},\ldots,\{\kappa-1,\kappa\}\}$. We identify the set of hyperedges of H with the set $\Sigma_3 \subseteq \{\{1,2,3\},\{1,2,4\},\ldots,\{\kappa-2,\kappa-1,\kappa\}\}$.

A *loading schedule for H* of length m is a list $S = (s_1,\ldots,s_m)$ of m elements such that the following three properties hold for all $t \in \{1,\ldots,m\}$: (1) $s_t \in \Sigma_1\cup\Sigma_2\cup\Sigma_3$; (2) if $s_t = \{i,j\}$, then there exist $t_1,t_2 \in \{1,\ldots,t-1\}$ such that $s_{t_1} = i$ and $s_{t_2} = j$; (3) if $s_t = \{i,j,k\}$, then there exist $t_1,t_2,t_3 \in \{1,\ldots,t-1\}$ such that $s_{t_1} = \{i,j\}$, $s_{t_2} = \{i,k\}$ and $s_{t_3} = \{j,k\}$. The loading schedule S is *valid* if no element of $\Sigma_1\cup\Sigma_2\cup\Sigma_3$ appears more than once and, for any $\{i,j,k\} \in \Sigma_3$, there exists an index $t \in \{1,\ldots,m\}$ such that $s_t = \{i,j,k\}$.

A *set of parameters for S* is a set of m integers defined as follows: for each $t \in \{1,\ldots,m\}$,

- if $s_t = i$, then the associated parameter is denoted by r_i and satisfies $r_i \in \{1,\ldots,n\}$;
- if $s_t = \{i,j\}$, then the associated parameter is denoted by f_{ij} and satisfies $f_{ij} \in \{1,\ldots,r_ir_j\}$;
- if $s_t = \{i,j,k\}$, then the associated parameter is denoted by e_{ijk} and satisfies $e_{ijk} \in \{1,\ldots,r_ir_jr_k\}$.

The set of parameters is *admissible* if there exists some constant $\gamma > 0$ such that all terms $\frac{n}{r_i}$, $\frac{r_ir_j}{f_{ij}}$, $\frac{f_{ij}}{r_i}$, $\frac{f_{ij}}{r_j}$, $\frac{f_{ij}f_{ik}}{r_ir_jr_k}$ and $\frac{f_{ij}f_{ik}f_{jk}/(r_ir_jr_k)}{e_{ijk}}$ are larger than n^γ. Our main result is as follows.

Theorem 1 *Let H be any constant-sized 3-uniform hypergraph. Let $S = (s_1,\ldots,s_m)$ be a valid loading schedule for H with an admissible set of parameters. There exists a quantum algorithm that, given as input a 3-uniform hypergraph G with n vertices, finds a sub-hypergraph of G isomorphic to H (and returns "no" if there are no such sub-hypergraphs) with probability at least some constant, and has query complexity $\tilde{O}\left(\mathsf{S} + \sum_{t=1}^m \left(\prod_{r=1}^t \frac{1}{\sqrt{\varepsilon_r}}\right)\frac{1}{\sqrt{\delta_t}}\mathsf{U_t}\right)$, where $\mathsf{S}, \mathsf{U}_t, \delta_t$, and ε_t are evaluated as follows: $\mathsf{S} = \sum_{\{i,j,k\}\in\Sigma_3} e_{ijk}$, and for each $t \in \{1,\ldots,m\}$,*

- *if $s_t = \{i\}$, then $\delta_t = \Omega(\frac{1}{r_i})$, $\varepsilon_t = \Omega(\frac{r_i}{n})$ and $\mathsf{U}_t = \tilde{O}\left(1 + \sum_{\{j,k\} \text{ such that } \{i,j,k\}\in\Sigma_3} \frac{e_{ijk}}{r_i}\right)$,*

- *if* $s_t=\{i,j\}$, *then* $\delta_t=\Omega(\frac{1}{f_{ij}})$, $\varepsilon_t=\Omega(\frac{f_{ij}}{r_ir_j})$ *and* $\mathsf{U}_t=\tilde{O}\left(1+\sum_{k \text{ such that } \{i,j,k\}\in\Sigma_3}\frac{e_{ijk}}{f_{ij}}\right)$,
- *if* $s_t=\{i,j,k\}$, *then* $\delta_t=\Omega(\frac{1}{e_{ijk}})$, $\varepsilon_t=\Omega(\frac{e_{ijk}r_ir_jr_k}{f_{ij}f_{ik}f_{jk}})$ *and* $\mathsf{U}_t=O(1)$.

The algorithm is based on the concept of m-level nested quantum walks. The quantum walk at the t-th level, for any $t\in\{1,\ldots,m\}$, corresponds to the element s_t of the loading schedule. The term S represents the setup cost of the whole nested walk. The term U_t represents the cost of updating the database of the t-th level walk. The terms ε_t and δ_t denote the spectral gap and the fraction of marked states, respectively, of the t-th level walk.

We then apply Theorem 1 to the case where H is the 4-clique, and optimize the parameters to obtain the following theorem.

Theorem 2 *There exists a quantum algorithm that finds the existence of a 4-clique in a 3-uniform hypergraph with high probability using* $\tilde{O}(n^{241/128})=O(n^{1.883})$ *queries.*

As another application, we can obtain a non-trivial upper bound of the query complexity of determining if a ternary operator is associative, but we omit it in this abstract due to space limitation (see arXiv:1310.4127v2).

Technical contribution. Roughly speaking, the subgraph finding algorithm by Lee, Magniez and Santha [12] works as follows. First, for each vertex i in the subgraph H that we want to find, a set V_i of randomly chosen vertices of the input graph is taken. This set V_i represents a set of candidates for the vertex i. Next, for each edge (i,j) in the subgraph H, a set of randomly chosen pairs from $V_i\times V_j$ is taken, representing a set of candidates for the edge (i,j). The most effective feature of their algorithm is to introduce a parameter for each ordered pair (V_i,V_j) that controls the average degree of a vertex in V_i toward V_j. This gives us more freedom for optimizing the algorithm than just taking all edges between randomly chosen subsets of V_i and V_j. To make the algorithm efficient, it is crucial to keep the degree of every vertex in V_i almost equal to the value specified by the parameter. For this, they carefully devise a procedure for taking pairs between V_i and V_j.

Our basic idea is similar in that we first, for each vertex i in the sub-hypergraph H that we want to find, take a set V_i of vertices of the input 3-uniform hypergraph as a set of candidates for i and then, for each hyperedge $\{i,j,k\}$ of H, take a random subset of triples in $V_i\times V_j\times V_k$. One may think that the remaining task is to fit the pair-taking procedure into the hypergraph case. It, however, turns out to be technically very complicated to generalize the pair-taking procedure from [12] to an efficient triple-taking procedure. Instead we cast the idea into the nested quantum walk of Jeffery, Kothari and Magniez [8] and employ probabilistic arguments. More concretely, we introduce a parameter that specifies the number e_{ijk} of triples to be taken from $V_i\times V_j\times V_k$ for each hyperedge $\{i,j,k\}$ of H. We then argue that, for randomly chosen e_{ijk} triples, the degree of each vertex sharply concentrates around its average, where the degree means the number of triples including the vertex (in this sense, the parameters e_{ijk} play essentially the same role as those of "average degrees" used in [8], but introducing e_{ijk} gives a neat formulation of the algorithm and this is effective particularly in handling such complicated cases as hypergraphs). This makes it substantially easier to analyze the complexity of all involved quantum walks, and enables us to completely analyze the complexity of our approach. Unfortunately, it turns out that this approach (taking the sets V_i first, and then e_{ijk} triples from each $V_i\times V_j\times V_k$) does not lead to any improvement over the naïve $O(n^2)$-query quantum algorithm.

Our key idea is to introduce, for each unordered pair $\{i,j\}$ of vertices in H, a parameter f_{ij}, and modify the approach as follows. After randomly choosing V_i,V_j,V_k, we take three random subsets $F_{ij}\subseteq V_i\times V_j$, $F_{jk}\subseteq V_j\times V_k$, and $F_{ik}\subseteq V_i\times V_k$ of size f_{ij}, f_{jk} and f_{ik}, respectively. We then randomly choose e_{ijk} triples from the set $\Gamma_{ijk}=\{(u,v,w)\mid (u,v)\in F_{ij},(u,w)\in F_{ik} \text{ and } (v,w)\in F_{jk}\}$. The difficulty here is that the size of Γ_{ijk} varies depending on the sets F_{ij}, F_{jk}, F_{ik}. Another difficulty is that, after taking many quantum-walks (i.e., performing the update operation many times), the distribution of the set of pairs can change. For these, we define the "marked states" (i.e., "absorbing states") of each level of the nested quantum walk as those that contain components (i.e., vertices, pairs or triples) of a copy of H inside the associated sets (i.e., V_i, F_{ij}, F_{jk}, F_{ik}, or Γ_{ijk}) *and satisfy certain regularity conditions*. We then show that the associated sets almost always satisfy the regularity conditions, by using concentration theorems for hypergeometric distributions. This regularity enables us to effectively bound the complexity of our new approach, giving in particular the claimed $\tilde{O}(n^{241/128})$-query upper bound when H is a 4-clique.

References

[1] Andris Ambainis. Quantum walk algorithm for element distinctness. *SIAM J. Comput.* 37(1): 210–239, 2007.

[2] Aleksandrs Belovs. Span programs for functions with constant-sized 1-certificates: extended abstract. In *Proceedings of STOC*, pages 77–84, 2012.

[3] Aleksandrs Belovs. Learning-graph-based quantum algorithm for k-distinctness. In *Proceedings of FOCS*, pages 207–216, 2012.

[4] Aleksandrs Belovs, Andrew M. Childs, Stacey Jeffery, Robin Kothari and Frédéric Magniez. Time-efficient quantum walks for 3-distinctness. In *Proceedings of ICALP, Part I*, pages 105–122, 2013.

[5] Aleksandrs Belovs and Ansis Rosmanis. On the power of non-adaptive learning graphs. In *Proceedings of CCC*, pages 44–55, 2013.

[6] Harry Buhrman, Christoph Dürr, Mark Heiligman, Peter Høyer, Frédéric Magniez, Miklos Santha and Ronald de Wolf. Quantum algorithms for element distinctness. *SIAM J. Comput.* 34(6): 1324–1330, 2005.

[7] Peter Høyer, Troy Lee and Robert Špalek. Negative weights make adversaries stronger. In *Proceedings of STOC*, pages 526–535, 2007.

[8] Stacey Jeffery, Robin Kothari and Frédéric Magniez. Nested quantum walks with quantum data structures. In *Proceedings of SODA*, pages 1474–1485, 2013.

[9] François Le Gall. Improved output-sensitive quantum algorithms for Boolean matrix multiplication. In *Proceedings of SODA*, pages 1464–1476, 2012.

[10] François Le Gall. Improved Quantum Algorithm for Triangle Finding via Combinatorial Arguments. In *Proceedings of FOCS*, pages 216–225, 2014.

[11] Troy Lee, Frédéric Magniez and Miklos Santha. Learning graph based quantum query algorithms for finding constant-size subgraphs. *Chicago J. Theor. Comput. Sci.* Article 10, 2012.

[12] Troy Lee, Frédéric Magniez and Miklos Santha. Improved quantum query algorithms for triangle finding and associativity testing. In *Proceedings of SODA*, pages 1486–1502, 2013.

[13] Frédéric Magniez, Ashwin Nayak, Jérémie Roland and Miklos Santha. Search via quantum walk. *SIAM J. Comput.* 40(1): 142–164, 2011.

[14] Frédéric Magniez, Miklos Santha and Mario Szegedy. Quantum algorithms for the triangle problem. *SIAM J. Comput.* 37(2): 413–424, 2007.

[15] Ben Reichardt. Span programs and quantum query complexity: The general adversary bound is nearly tight for every Boolean function. In *Proceedings of FOCS*, pages 544–551, 2009.

[16] Ben Reichardt. Reflections for quantum query algorithms. In *Proceedings of SODA*, pages 560–569, 2011.

[17] Robert Špalek and Mario Szegedy. All quantum adversary methods are equivalent. *Theory of Computing* 2(1): 1–18, 2006.

[18] Mario Szegedy. Quantum speed-up of Markov chain based algorithms. In *Proceedings of FOCS*, pages 32–41, 2004.

[19] Virginia Vassilevska Williams and Ryan Williams. Finding, minimizing, and counting weighted subgraphs. *SIAM J. Comput.* 42(3): 831–854, 2013.

[20] Virginia Vassilevska Williams and Ryan Williams. Subcubic equivalences between path, matrix and triangle problems. In *Proceedings of FOCS*, pages 645–654, 2010.

[21] Ryan Williams. A new algorithm for optimal 2-constraint satisfaction and its implications. *Theor. Comput. Sci.* 348(2-3): 357–365, 2005.

[22] Ryan Williams. *Algorithms and resource requirements for fundamental problems*. Ph.D. Thesis, Carnegie Mellon University, 2007.

[23] Yechao Zhu. Quantum query complexity of constant-sized subgraph containment. *Int. J. Quant. Inf.* 10(3): 1250019, 2012.