Merkle Puzzles in a Quantum World

Gilles Brassard[1], Peter Høyer[2], Kassem Kalach[1],
Marc Kaplan[1], Sophie Laplante[3], and Louis Salvail[1]

[1] Département d'informatique et de recherche opérationnelle
Université de Montréal, C.P. 6128, Succursale Centre-ville
Montréal (QC), H3C 3J7 Canada
[2] Department of Computer Science, University of Calgary
2500 University Drive N.W., Calgary, AB, T2N 1N4 Canada
[3] LRI, Université Paris-Sud, 91400 Orsay, France

Abstract. In 1974, Ralph Merkle proposed the first unclassified scheme for secure communications over insecure channels. When legitimate communicating parties are willing to spend an amount of computational effort proportional to some parameter N, an eavesdropper cannot break into their communication without spending a time proportional to N^2, which is quadratically more than the legitimate effort. We showed in an earlier paper that Merkle's schemes are completely insecure against a quantum adversary, but that their security can be partially restored if the legitimate parties are also allowed to use quantum computation: the eavesdropper needed to spend a time proportional to $N^{3/2}$ to break our earlier quantum scheme. Furthermore, all previous *classical* schemes could be broken completely by the onslaught of a quantum eavesdropper and we conjectured that this is unavoidable.

We give two novel key agreement schemes in the spirit of Merkle's. The first one can be broken by a quantum adversary that makes an effort proportional to $N^{5/3}$ to implement a quantum random walk in a Johnson graph reminiscent of Andris Ambainis' quantum algorithm for the element distinctness problem. This attack is optimal up to logarithmic factors. Our second scheme is purely classical, yet it cannot be broken by a quantum eavesdropper who is only willing to expend effort proportional to that of the legitimate parties.

Keywords: Merkle Puzzles, Public Key Distribution, Quantum Cryptography.

1 Introduction

While Ralph Merkle was delivering the 2005 International Association for Cryptologic Research (IACR) Distinguished Lecture at the CRYPTO annual conference in Santa Barbara, describing his original unpublished 1974 scheme [15] for public key distribution (much simpler and more elegant than his subsequently published, yet better known, Merkle Puzzles [16]), one of us (Brassard) immediately realized that this scheme was totally insecure against an eavesdropper

equipped with a quantum computer. The obvious question was: can Merkle's idea be repaired and made secure again in our quantum world? The defining characteristics of Merkle's protocol are that (1) the legitimate parties communicate strictly through an authenticated classical channel on which eavesdropping is unrestricted and (2) a protocol is deemed to be *secure* if the cryptanalytic effort required of the eavesdropper to learn the key exchanged by the legitimate parties grows super-linearly with the legitimate work.

We partially repaired Merkle's scheme in Ref. [8] with a scheme in which the eavesdropper needed an amount of work in $\Omega(N^{3/2})$ to obtain the key established by quantum legitimate parties whose amount of work is in $O(N)$. This was not quite as good as the work in $\Omega(N^2)$ required by a *classical* eavesdropper against Merkle's original scheme, but significantly better than the work in $O(N)$ sufficient for a *quantum* eavesdropper against the same scheme. Two main questions were left open in Ref. [8]:

1. Can the quadratic security possible in a classical world be restored in our quantum world?
2. Is any security possible at all if the legitimate parties are purely classical, yet the eavesdropper is endowed with a quantum computer?

We give two novel key distribution protocols to address these issues. In the first protocol, the legitimate parties use quantum computers and classical authenticated communication to establish a shared key after $O(N)$ expected queries to two black-box random functions (which can be modelled with a single random oracle). We then give a nontrivial quantum cryptanalytic attack that uses a quantum random walk in a Johnson graph, much like Andris Ambainis' algorithm to solve the element distinctness problem [2], which allows a quantum eavesdropper to learn the key after $\Theta(N^{5/3})$ queries to the functions. Finally, we prove that our attack is optimal up to logarithmic factors. Therefore, we have not quite restored the quadratic security possible in a classical world, but we have made significant progress towards it.

Second, we give a *purely classical* protocol, in which the legitimate parties use classical communication and classical computation to establish a key after $O(N)$ calls to similar black-box random functions. We then attack this protocol with a quantum cryptanalytic algorithm that uses $\Theta(N^{13/12})$ queries to the functions. As unlikely as it may sound, this attack is optimal (up to logarithmic factors) and therefore it is not possible to break this purely classical protocol with a quantum attack that uses an amount of resource linear in the legitimate effort.

Before describing our new protocols (Sects. 3 and 4), quantum attacks against them (Sects. 3.1 and 4.1), proofs of optimality for those attacks (Sects. 3.2 and 4.2), and conjectures about the existence of even better schemes (Sect. 5), we begin with a review (lifted from Ref. [8]) of Merkle's original idea, its meltdown against a quantum eavesdropper, and our earlier partial quantum solution (Sect. 2). Some of the technical tools required by our quantum attacks are reviewed in the Appendix and new lower bound techniques are introduced.

2 Merkle's Original Scheme and How to Break and Partially Repair It

The first unclassified document ever written that pioneered public key distribution and public key cryptography was a project proposal written in 1974 by Merkle when he was a student in Lance Hoffman's CS244 course on Computer Security at the University of California, Berkeley [15]. Hoffman rejected the proposal and Merkle dropped the course but "kept working on the idea" and eventually published it as one of the most seminal cryptographic papers in the second half of the twentieth century [16]. Merkle's scheme in his published paper was somewhat different from his original 1974 idea, but both share the property that they "force any enemy to expend an amount of work which increases as the square of the work required of the two [legitimate] communicants" [16]. It took 35 years before Boaz Barak and Mohammad Mahmoody-Ghidary proved that this quadratic discrepancy between the legitimate and eavesdropping efforts are the best possible in a classical world [3].

In his IACR Distinguished Lecture [4], which he delivered at the CRYPTO '05 Conference in Santa Barbara, Merkle described from memory his first solution to the problem of secure communications over insecure channels. As a wondrous coincidence, he unsuspectingly opened up a box of old folders a mere three weeks after his Lecture and happily recovered his long-lost CS244 Project Proposal, together with comments handwritten by Hoffman [15]! To quote his original typewritten words:

```
Method 1:    Guessing.  Both sites guess at keywords.  These
             guesses are one-way encrypted, and transmitted to the
             other site.  If both sites should chance to guess at
             the same keyword, this fact will be discovered when
             the encrypted versions are compared, and this keyword
             will then be used to establish a communications link.
Discussion:  No, I am not joking.
```

In more modern terms, let f be a one-way permutation. In order to "one-way encrypt" x, as Merkle said in 1974, we assume that one can compute $f(x)$ in unit time for any given input x but that the only way to retrieve x given $f(x)$ is to try preimages and compute f on them until one is found that maps to $f(x)$. This is known as the *black-box* (or *oracle*) model. Hereinafter, in accordance with this model, efficiency is defined *solely* in terms of the number of calls to such black-box functions (there could be more than one). In the quantum case, these calls can be made in superposition of inputs. We also assume throughout this paper (as did Merkle) that an authenticated channel is available between the legitimate communicants, although this channel offers no protection against eavesdropping.

The "keywords" guessed at by "both sites" are random points in the domain of f. They are "one-way encrypted" by applying f to them. If there are N^2

[4] `www.iacr.org/publications/dl`.

points in the domain of f, it suffices to guess $O(N)$ keywords at each site before a variation on the birthday paradox makes it overwhelmingly likely that "both sites should chance to guess at the same keyword", which becomes their shared key. An eavesdropper who listens to the entire conversation has no other way to obtain this key than to invert f on the revealed common encrypted keyword. In accordance with the black-box model, this can only be done by trying on the average half the points in the domain of f before one is found that is mapped by f to the target value. This will require an expected number of calls to f in $\Omega(N^2)$, which is quadratic in the legitimate effort.

Shortly thereafter, Whitfield Diffie and Martin Hellman discovered a celebrated method for public-key distribution that makes the cryptanalytic effort *apparently* exponentially harder than the legitimate effort [10]. However, no proof is known that the Diffie-Hellman scheme is secure at all since it relies on the conjectured difficulty of extracting discrete logarithms, an assumption doomed to fail whenever quantum computers become available. In contrast, Merkle's approach offers provable quadratic security against any possible classical attack, under the sole assumption that f cannot be inverted by any other means than exhaustive search.

Next, we explain why Merkle's original proposal becomes completely insecure if the eavesdropper is capable of quantum computation (Merkle's published "puzzles" [16] are equally insecure). We then sketch a protocol from Ref. [8] that is not completely broken. This is be achieved by granting similar quantum computation capabilities to one of the legitimate communicating parties.

2.1 Quantum Attack and Partial Remedy

Let us now assume that function f can be computed quantum mechanically on a superposition of inputs. In this case, Merkle's original scheme is completely compromised by way of Grover's algorithm [11]. Indeed, this algorithm needs only $O(\sqrt{N^2}) = O(N)$ calls on f in order to invert it on any given point of its image, making the cryptanalytic task as easy (up to constant factors) as the legitimate key setup process. [5]

To remedy the situation, we allow the communicating parties to use quantum computers as well (actually, one of the parties will remain classical), and we increase the domain of f from N^2 to N^3 points. Instead of having both sites transmit one-way encrypted guesses to the other site, one site called Alice chooses N distinct random values $x_1, x_2, \ldots, x_N$ and transmits them, one-way encrypted by the application of f, to the other site called Bob. Let $Y = \{f(x_i) \mid 1 \leqslant i \leqslant N\}$

[5] If an unstructured search problem has t solutions among M candidates, Grover's algorithm [11], or more precisely its so-called BBHT generalization [6], can find one of the solutions after $O(\sqrt{M/t})$ *expected* calls to a function that recognizes solutions among candidates. However, Theorem 4 of Ref. [7] implies that, whenever the number $t > 0$ is known, a solution can be found *with certainty* after $O(\sqrt{M/t})$ calls to that function in the *worst case*. From now on, when we mention Grover's algorithm or BBHT, we really mean this improvement according to Ref. [7].

denote the set of encrypted keywords received by Bob, which becomes known to the eavesdropper. Now, Bob defines Boolean function g on the same domain as f by

$$g(x) = \begin{cases} 1 \text{ if } f(x) \in Y \\ 0 \text{ otherwise}. \end{cases}$$

Out of N^3 points in the domain of f, there are exactly $t = N$ solutions to the problem of finding an x so that $g(x) = 1$. It suffices for Bob to apply the BBHT generalization [6] of Grover's algorithm [11], which finds such an x after $O(\sqrt{N^3/t}\,) = O(\sqrt{N^2}\,) = O(N)$ calls on g (and therefore on f). Bob sends back $f(x)$ to Alice, who knows the value of x because she was careful to keep her randomly chosen points. Therefore, it suffices of $O(N)$ calls on f by Alice and Bob for them to agree on key x.[6]

The eavesdropper, on the other hand, is faced with the need to invert f on a specific point of its image. Even with a quantum computer, this requires a number of calls on f proportional to the square root of the number of points in its domain [5], which is $\Omega(\sqrt{N^3}\,) = \Omega(N^{3/2})$. This is more effort than what is required of the legitimate parties, yet less than quadratically so, as would have been possible in a classical world. Even though we have avoided the meltdown of Merkle's original approach, the introduction of quantum computers available to all sides seems to be to the advantage of the codebreakers. Can we remedy this situation? Furthermore, is any security possible at all against a quantum computer if both legitimate parties are restricted to being purely classical? We address these two questions in the rest of this paper.

3 Improved Key Distribution Scheme

For any positive integer N, let $[N]$ denote the set of integers from 1 to N. We describe our novel key distribution protocol assuming the existence of *two* black-box random functions $f : [N^3] \to [N^k]$ and $g : [N^3] \times [N^3] \to [N^{k'}]$ that can be accessed in quantum superposition of inputs. Constants k and k' are chosen large enough so that there is no collision in the images of f and g, except with negligible probability. (For simplicity, we shall systematically disregard the possibility that such collisions might exist.) Notice that *a single binary random oracle* (which "implements" a random function from the integers to $\{0, 1\}$) could be used to define both functions f and g provided we disregard logarithmic factors in our analyses since $O(\log N)$ calls to the random oracle would suffice to compute f or g on any single input. For this reason, it is understood hereinafter that all our results are implicitly stated "up to logarithmic factors". As mentioned in the previous section, the only resource that we consider in our analyses

[6] As we made clear already, we are only concerned in this paper by the number of calls made to black-box functions. Nevertheless, *if* we cared also about computational efficiency, Bob would sort the elements of Y in increasing order after receiving them from Alice so that he can quickly determine, given any $y = f(x)$, whether or not $y \in Y$, which is needed to compute function g.

of efficiency and lower bounds is the number of calls made to these functions or, equivalently, to the underlying binary random oracle.

Protocol 1.

1. Alice picks at random N distinct values $\{x_i\}_{i=1}^{N}$ with $x_i \in [N^3]$ and transmits the encrypted values $y_i = f(x_i)$ to Bob. Let X and Y denote $\{x_i \mid 1 \leqslant i \leqslant N\}$ and $\{y_i \mid 1 \leqslant i \leqslant N\}$, respectively. Note that Alice knows both X and Y, whereas Bob and the eavesdropper have immediate knowledge (i.e. without querying the black-box for function f) of Y only.
2. Bob finds the pre-images x and x' of *two* distinct random elements in Y. To find each one of them, he uses BBHT [6] to search for an x such that $\phi(x) = 1$, where $\phi : [N^3] \to \{0, 1\}$ is defined as follows:

$$\phi(x) = \begin{cases} 1 \text{ if } f(x) \in Y \\ 0 \text{ otherwise}\,. \end{cases}$$

   There are exactly N values of x such that $\phi(x) = 1$, out of N^3 points in the domain of ϕ. Therefore, Bob can find one such random x with $O(\sqrt{N^3/N}) = O(N)$ calls to function f. He needs to repeat this process twice in order to get both x and x'. (A small variation in function ϕ can be used the second time to make sure that $x' \neq x$).
3. Bob sends back $w = g(x, x')$ to Alice.
4. Because Alice had kept her randomly chosen set X, there are only N^2 candidate pairs $(x_i, x_j) \in X \times X$ such that $g(x_i, x_j)$ could equal w. Using Grover's algorithm, she can find the one pair (x, x') that Bob has in mind with $O(\sqrt{N^2}) = O(N)$ calls to function g.
5. The key shared by Alice and Bob is the pair (x, x').

All counted, Alice makes N calls to f in step 1 and $O(N)$ calls to g in step 4, whereas Bob makes $O(N)$ calls to f in step 2 and a single call to g in step 3. If the protocol is constructed over a binary random oracle, it will have to be called $O(N \log N)$ times since it takes $O(\log N)$ binary queries to compute either function on any given input.

3.1 Quantum Attack

All the obvious (and not so obvious) cryptanalytic attacks against this scheme, such as direct use of Grover's algorithm (or BBHT), or even more sophisticated attacks based on amplitude amplification [7], require the eavesdropper to call $\Omega(N^2)$ times functions f and/or g. Unfortunately, a more powerful attack based on the more recent paradigm of quantum walks in Markov chains [17] allows the eavesdropper to recover Alice and Bob's key (x, x') with an expected $O(N^{5/3})$ calls to f and $O(N)$ calls to g. This attack was inspired by Ambainis' quantum algorithm for element distinctness [2], which can find the unique pair (i, j) such that $c(i) = c(j)$ with $O(N^{2/3})$ expected queries to single-collision function c whose domain contains N elements (whereas all previous approaches based on Grover's algorithm and amplitude amplification [12, 9] had required $\Omega(N^{3/4})$ queries).

Theorem 1. *There exists an eavesdropping strategy that outputs the pair (x, x') in Protocol 1 with $O(N^{5/3})$ expected quantum queries to functions f and g.*

Proof. In a nutshell, we apply Ambainis' algorithm for element distinctness with two modifications: (1) instead of looking for i and j such that $c(i) = c(j)$, we are looking for x and x' such that $g(x, x') = w$ and (2) instead of being able to get randomly chosen values in the image of h with a single call to oracle h per value, we need to get random elements of X by applying BBHT on the list Y, which requires $O(\sqrt{N^3/N}) = O(N)$ calls to oracle f per element. The second modification explains why the number of calls to f, compared to $O(N^{2/3})$ calls to c for element distinctness, is multiplied by $O(N)$. Hence, we need $O(N^{5/3})$ calls to function f. To determine the number of calls required to function g, however, we have to delve deeper into the eavesdropping algorithm.

The eavesdropping algorithm uses a quantum walk on a Johnson graph—see the Appendix for a review of this topic. Each node of the graph contains some number r (to be determined later) of distinct elements of X. We are looking for a node that contains the two elements x and x' such that $g(x, x') = w$, where w is the value announced by Bob in step 3 of the protocol. We apply Theorem 5 (Appendix) to analyse the cost of a quantum walk on this graph [2, 17]. The set up cost S corresponds to finding r random elements of X. Since BBHT can be used to find one such element with $O(N)$ calls to f, and even to find an element of X guaranteed to be different from those already in the initial node (provided $k \ll N$, which it will be), $\mathsf{S} = O(rN)$ calls to f. The update cost U corresponds to finding one random element of X not already in the node, which is $\mathsf{U} = O(N)$ calls to f, again by BBHT. The checking cost C requires us to decide if there is a pair (x, x') of elements in the node such that $g(x, x') = w$, which can be done with $O(\sqrt{r^2}) = O(r)$ calls to g using Grover's algorithm since there are r^2 pairs of elements in the node. Putting it all together, the expected cryptanalytic cost is

$$\begin{aligned}
&\mathsf{S} + O\left(\tfrac{N}{r}(\sqrt{r}\,\mathsf{U} + \mathsf{C})\right) \\
&= O\left((rN \text{ calls to } f) + \tfrac{N}{r}\left(\sqrt{r}(N \text{ calls to } f) + (r \text{ calls to } g)\right)\right) \\
&= O\left(rN + N^2/\sqrt{r}\right) \text{ calls to } f \textbf{ and } O(N) \text{ calls to } g\,.
\end{aligned}$$

To minimize the number of calls to f, we choose r so that $rN = N^2/\sqrt{r}$, which is $r = N^{2/3}$. It follows that a quantum eavesdropper is able to find the key (x, x') with an expected $O(rN) = O(N^{5/3})$ calls to f and $O(N)$ calls to g. □

Note that the use of Grover's algorithm in the checking step was not necessary to prove Theorem 1. Should this step be carried out classically, this would result in $\mathsf{C} = O(r^2)$ calls to g. The net result would be that the key is found after an expected $O(N^{5/3})$ calls to f and also $O(N^{5/3})$ calls to g.

3.2 Lower Bound

The proof that the quantum attack described above against our protocol is optimal proceeds in three steps.

1. We define a search problem reminiscent of element distinctness;
2. We prove a lower bound on the difficulty to solve this search problem; and
3. We reduce this search problem to the eavesdropping problem against our protocol. More precisely, we show that any attack on our key distribution scheme that would have a nonvanishing probability of success after $o(N^{5/3})$ calls to functions f and g could be turned into an algorithm capable of solving the search problem more efficiently than possible.

First, consider a function $c : [N] \to [N]$ so that there exists a single pair (i, j), $1 \leqslant i < j \leqslant N$, for which $c(i) = c(j)$. Ambainis' quantum algorithm for element distinctness [2] can find this pair with $O(N^{2/3})$ queries to function c and Scott Aaronson and Yaoyun Shi proved that this is optimal even for the decision version of this problem [1].

Now, consider a function $h : [N] \times [N^2] \to [N]'$, where $[N]'$ denotes $\{0\} \cup [N]$. The domain of this function is composed of N "buckets" of size N^2, where $h(i, \cdot)$ corresponds to the i^{th} bucket, $1 \leqslant i \leqslant N$. In bucket i, all values of the function are 0 except for one single random $v_i \in [N^2]$ for which $h(i, v_i) = c(i)$:

$$h(i,j) = \begin{cases} c(i) & \text{if } j = v_i \\ 0 & \text{otherwise}\,. \end{cases}$$

It follows from the definitions of c and h that there is a single pair of distinct a and b in the domain of h such that $h(a) = h(b) \neq 0$. How difficult is it to find this pair given a black box for function h but no direct access to c?

Lemma 1. *Given h structured as above, finding the pair of distinct elements a and b in the domain of h such that $h(a) = h(b) \neq 0$ requires $\Omega(N^{5/3})$ quantum queries to h, except with vanishing probability.*

Proof. This problem can be modelled as the composition of element distinctness across buckets with finding the single non-zero entry in each bucket. It is therefore a special case of technical Lemma 5, stated in the Appendix, with parameters $\alpha = N$ (the number of buckets) and $\beta = N^2$ (the size of the buckets). It follows that finding the desired pair (a, b) requires

$$\Omega(\alpha^{2/3}\beta^{1/2}) = \Omega(N^{2/3}\sqrt{N^2}\,) = \Omega(N^{5/3})$$

quantum queries to h, except with vanishing probability. □

Consider now a slightly different search problem in which there are no buckets anymore, but there is an added coordinate in the image of the function: $h' : [N^3] \to [N]' \times [N]'$ is defined so that $h'(a) = (0, 0)$ on all but N randomly chosen points in its domain, namely w_1, w_2,..., w_N. On these N points, $h'(w_i) = (i, c(i))$, where c is the function considered at the beginning of this section. We are required to find the unique pair of distinct a and b in $[N^3]$ such that $\pi_2(h'(a)) = \pi_2(h'(b)) \neq 0$, where "$\pi_2$" denotes the projection on the second coordinate (similarly for "π_1"). The lower bound on the earlier search problem

concerning h implies directly the same lower bound on the new search problem concerning h' since any algorithm capable of solving the new problem can be used at the same cost to solve the earlier problem through randomization. In other words, the more structured version of the problem cannot be harder than the less structured one. The next Lemma formalizes the argument above.

Lemma 2. *Given h' structured as above, finding the pair of distinct elements a and b in the domain of h' such that $\pi_2(h'(a)) = \pi_2(h'(b)) \neq 0$ requires $\Omega(N^{5/3})$ quantum queries to h', except with vanishing probability.*

Proof. Define intermediary function $\tilde{h} : [N] \times [N^2] \to [N]' \times [N]'$ by

$$\tilde{h}(i,j) = \begin{cases} (i, h(i,j)) = (i, c(i)) & \text{if } h(i,j) \neq 0 \\ (0, h(i,j)) = (0,0) & \text{otherwise}\,. \end{cases}$$

It is elementary to reduce the search problem concerning h to the one concerning $\tilde{h}$ as well as the search problem concerning $\tilde{h}$ to the one concerning h'. Therefore, the lower bound concerning h given by Lemma 1 applies *mutatis mutandis* to h'. □

Finally, we show how to reduce the search problem concerning h' to the cryptanalytic difficulty for the eavesdropper to determine the key that Alice and Bob have established by using our protocol. This is the last step in proving the security of our scheme.

Theorem 2. *Any eavesdropping strategy that recovers the key (x, x') in protocol 1 requires a total of $\Omega(N^{5/3})$ quantum queries to functions f and g, except with vanishing probability.*

Proof. Consider any eavesdropping strategy $\mathcal{A}$ that listens to the communication between Alice and Bob and tries to determine the key (x, x') by querying black-box functions f and g. In fact, there are no Alice and Bob at all! Instead, there is a function $h' : [N^3] \to [N]' \times [N]'$ as described above, for which we want to solve the search problem by using unsuspecting $\mathcal{A}$ as a resource.

We start by supplying $\mathcal{A}$ with a completely fake "conversation" between "Alice" and "Bob": for sufficiently large k and k', we choose randomly N points $y_1, y_2, \ldots, y_N$ in $[N^k]$ and one point $w \in [N^{k'}]$ and we pretend that Alice has sent the y's to Bob and that Bob has responded with w. We also choose random functions $\hat{f} : [N^3] \to [N^k]$ and $\hat{g} : [N^3] \times [N^3] \to [N^{k'}]$, as well as a random Boolean $s \in \{\mathsf{true}, \mathsf{false}\}$. Note that the selection of $\hat{f}$ and $\hat{g}$ may take a lot of *time*, but this does not count towards the number of *queries* that will be made of function h', and our lower bound on the search problem concerns *only* this number of queries. The Boolean s indicates, when true (resp. false), that the fake "execution" is such that "Bob" has first picked x and then x' such that $x < x'$ (resp. $x' > x$). Both cases happen with probability $1/2$ in any real execution and for any public announcements Y and w. The value s will be used in the reduction to distinguish between $g(x, x')$ and $g(x', x)$ so that only $g(x, x')$ will be set to w.

Now, we wait for $\mathcal{A}$'s queries to f and g.

- When $\mathcal{A}$ asks for $f(i)$ for some $i \in [N^3]$, there are two possibilities.
  - If $h'(i) = (0,0)$, return $\hat{f}(i)$ to $\mathcal{A}$ as value for $f(i)$.
  - Otherwise, return $y_{\pi_1(h'(i))}$.
- When $\mathcal{A}$ asks for $g(i,j)$ for some $i,j \in [N^3]$, there are again two possibilities.
  - If $\pi_2(h'(i)) = \pi_2(h'(j)) \neq 0$ and either s is true and $i < j$ or s is false and $i > j$, return w as value for $g(i,j)$.
  - Otherwise, return $\hat{g}(i,j)$.

Suppose $\mathcal{A}$ happily returns the pair (i,j) for which it was told that $g(i,j) = w$, which is what a successful eavesdropper is supposed to do. This pair is in fact the answer to the search problem concerning h' since $g(i,j) = w$ implies that $\pi_2(h'(i)) = \pi_2(h'(j)) \neq 0$, except with the negligible probability that $\hat{g}(i',j') = w$ for some query (i',j') that $\mathcal{A}$ asks about g.

Queries asked by $\mathcal{A}$ concerning f and g are answered in the same way as they would be if f and g were two random functions consistent with the Y and w announced by Alice and Bob during the execution of a real protocol. To see this, remember that Y (subset of $[N^k]$) and w (element of $[N^{k'}]$) are uniformly picked at random in both the simulated and the real worlds. Moreover, the simulated function f is such that $f(i)$ is random when $h'(i) = (0,0)$. The remaining N output values are in Y, as expected by $\mathcal{A}$. On the other hand, the simulated function g is random everywhere except for one single input pair (i,j), $i \neq j$ for which $g(i,j) = w$, as it is also expected by $\mathcal{A}$. Therefore, $\mathcal{A}$ will behave in the environment provided by the simulation exactly as in the real world. Since we disregard the negligible possibility that g might not be be one-to-one, the reduction solves the search problem concerning h' whenever $\mathcal{A}$ succeeds in finding the key. Notice finally that each (new) question asked by $\mathcal{A}$ to either f or g translates to one or two questions actually asked to h'.

It follows that any successful cryptanalytic strategy that makes $o(N^{5/3})$ total queries to f and g would solve the search problem with only $o(N^{5/3})$ queries to function h', which is impossible, except with vanishing probability. This establishes the $\Omega(N^{5/3})$ lower bound on the cryptanalytic difficulty of breaking our key exchange protocol, again except with vanishing probability, which matches the upper bound provided by the explicit attack given in Sect. 3.1. □

4 Fully Classical Key Distribution Scheme

In this section, we revert to the original setting imagined by Merkle in the sense that Alice and Bob are now purely classical. However, we allow full quantum power to the eavesdropper. Recall that Merkle's original schemes [15, 16] are completely broken in this context [8]. Is it possible to restore *some* security in this highly adversarial (and unfair!) scenario? The following purely classical key distribution protocol, which is inspired by our quantum protocol described in the previous section, provides a positive answer to this conundrum.

This time, black-box random functions f and g are defined on a smaller domain to compensate for the fact that classical Alice and Bob can no longer use Grover's algorithm. Specifically, $f : [N^2] \to [N^k]$ and $g : [N^2] \times [N^2] \to [N^{k'}]$, again with sufficiently large k and k' to avoid collisions in these functions, except with negligible probability (k and k' need not be the same here as in the previous section). As before, these two functions could be replaced by a single binary random oracle. For simplicity, we choose N to be a perfect square.

Protocol 2.

1. Alice picks at random N distinct values $\{x_i\}_{i=1}^{N}$ with $x_i \in [N^2]$ and transmits the encrypted values $y_i = f(x_i)$ to Bob. Let X and Y denote $\{x_i \mid 1 \leqslant i \leqslant N\}$ and $\{y_i \mid 1 \leqslant i \leqslant N\}$, respectively.
2. Bob finds the pre-images x and x' of two distinct random elements in Y. To find each one of them, he chooses random values in $[N^2]$ and applies f to them until one is found whose image is in Y. By virtue of the birthday paradox, he is expected to succeed after $O(\sqrt{N^2}) = O(N)$ calls to function f. *Until now this is identical to Merkle's original scheme*, except for the fact that Bob needs to find two elements of X rather than one.
3. Bob sends back $w = g(x, x')$ to Alice. In addition, he chooses $\sqrt{N} - 2$ random elements from $Y \setminus \{f(x), f(x')\}$ and he forms a set Y' of cardinality $\sqrt{N}$ by adding $f(x)$ and $f(x')$ to those elements. He sends the elements of Y' to Alice in increasing order of values.
4. Because Alice had kept her randomly chosen set X, she knows the preimages of each element of Y'. Let X' denote $\{x \in X \mid f(x) \in Y'\}$. By exhaustive search over all pairs of elements of X', Alice finds the one pair (x, x') such that $g(x, x') = w$.
5. The key shared by Alice and Bob is the pair (x, x').

All counted, Alice makes N calls to f in step 1 and at most N calls to g in step 4 because there are $\sqrt{N}\sqrt{N} = N$ pairs of elements of X' and one of them is the correct one. As for Bob, he makes an expected $O(N)$ calls to f in step 2 and a singe call to g in step 3. The total expected number of calls to f and g is therefore in $O(N)$ for both legitimate parties.

4.1 Quantum Attack

Theorem 3. *There exists an eavesdropping strategy that outputs the pair (x, x') in Protocol 2 with $O(N^{13/12})$ expected quantum queries to functions f and g.*

Proof. A quantum eavesdropper can set up a walk in a Johnson graph very similar to the one explained in Sect. 3.1, except that now the nodes in the graph contain some number r (to be determined later) of distinct elements of X' (rather than of X). The eavesdropper can find random elements of X' from his knowledge of Y' with an expected

$$O\left(\sqrt{N^2/\sqrt{N}}\right) = O\big(N^{3/4}\big)$$

calls to f per element of X'. Therefore, $\mathsf{S} = O(rN^{3/4})$ calls to f, $\mathsf{U} = O(N^{3/4})$ calls to f and $\mathsf{C} = O(r)$ calls to g. Furthermore, δ is still $\Theta(1/r)$ but $\varepsilon = \Omega(r^2/N)$.

Putting it all together, the expected quantum cryptanalytic cost is

$$\begin{aligned}
&\mathsf{S} + O\left(\tfrac{\sqrt{N}}{r}(\sqrt{r}\,\mathsf{U} + \mathsf{C})\right) \\
&= O\left((rN^{3/4} \text{ calls to } f) + \tfrac{\sqrt{N}}{r}\left(\sqrt{r}(N^{3/4} \text{ calls to } f) + (r \text{ calls to } g)\right)\right) \\
&= O\left(rN^{3/4} + N^{5/4}/\sqrt{r}\right) \text{ calls to } f \;\; \textbf{and} \;\; O(\sqrt{N}\,) \text{ calls to } g\,.
\end{aligned}$$

To minimize the number of calls to f, we choose r so that $rN^{3/4} = N^{5/4}/\sqrt{r}$, which is $r = N^{1/3}$. It follows that a quantum eavesdropper is able to find the key (x, x') with an expected $O(rN^{3/4}) = O(N^{13/12})$ calls to f and $O(\sqrt{N})$ calls to g. □

4.2 Lower Bound

The proof that it is not possible to find the key (x, x') with fewer than $\Omega(N^{13/12})$ calls to f and g, except with vanishing probability, follows the same lines as the lower bound proof in Sect. 3.2. It is therefore possible for purely classical Alice and Bob to agree on a shared key after calling f and g an expected number of times in the order of N whereas it is not possible, even for a *quantum* eavesdropper, to be privy of their secret with an effort in the same order, except with vanishing probability.

We refer the reader to Sect. 3 for the meaning of notation $[N]$ and to Sect. 3.2 for the definitions of projectors π_1, π_2, and the meaning of notation $[N]'$.

Consider a function $c : [\sqrt{N}\,] \to [\sqrt{N}\,]$ so that there is a single pair (i, j), $1 \leqslant i < j \leqslant \sqrt{N}$, for which $c(i) = c(j)$. Aaronson and Shi's lower bound [1] tells us that finding this pair requires $\Omega((\sqrt{N})^{2/3}) = \Omega(N^{1/3})$ calls to function c. Now, consider a function $h : [\sqrt{N}\,] \times [N^{3/2}] \to [\sqrt{N}\,]'$ where $h(i, \cdot)$ denotes the i^{th} bucket, $1 \leqslant i \leqslant \sqrt{N}$. In bucket i, all values of the function are 0 except for one: there is a single random $v_i \in [N^{3/2}]$ such that $h(i, v_i) = c(i)$. It follows from the definitions of c and h that there is a single pair of distinct a and b in the domain of h such that $h(a) = h(b) \neq 0$.

Lemma 3. *Given h structured as above, finding the pair of distinct elements a and b in the domain of h such that $h(a) = h(b) \neq 0$ requires $\Omega(N^{13/12})$ quantum queries to h, except with vanishing probability.*

Proof. The proof is identical to the one for Lemma 1, *mutatis mutandis*. It is again a special case of Lemma 5, but with parameters $\alpha = \sqrt{N}$ (the number of buckets) and $\beta = N^{3/2}$ (the size of the buckets). It follows that finding the desired pair (a, b) requires

$$\Omega(\alpha^{2/3}\beta^{1/2}) = \Omega\Big(\sqrt{N}^{\,2/3}\sqrt{N^{3/2}}\Big) = \Omega(N^{13/12})$$

quantum queries to h, except with vanishing probability. □

Let $h' : [N^2] \to [\sqrt{N}\,]' \times [\sqrt{N}\,]'$ denote the unstructured version of the same search problem for h, defined the same way as in Sect. 3.2, *mutatis mutandis*. There is a single pair of distinct elements a and b such that $\pi_2(h'(a)) = \pi_2(h'(b)) \neq 0$. The problem of finding this pair is at least as difficult as finding the collision in h.

Lemma 4. *Given h' structured as above, finding the pair of distinct elements a and b in the domain of h' such that $\pi_2(h'(a)) = \pi_2(h'(b)) \neq 0$ requires $\Omega(N^{13/12})$ quantum queries to h', except with vanishing probability.*

It remains to show that the search problem concerning h' reduces to the cryptanalytic difficulty for the eavesdropper to determine the key established by Alice and Bob.

Theorem 4. *Any eavesdropping strategy that recovers the key (x, x') in protocol 2 requires a total of $\Omega(N^{13/12})$ quantum queries to functions f and g, except with vanishing probability.*

Proof. Consider any eavesdropping strategy $\mathcal{A}$ that listens to the communication between Alice and Bob and tries to determine the key (x, x') by querying the black-box functions f and g. As before, the reduction does not have access to Alice and Bob but instead, to a function $h' : [N^2] \to [\sqrt{N}\,]' \times [\sqrt{N}\,]'$ as described above and given as an oracle, for which we want to solve the search problem by using $\mathcal{A}$ as a resource.

We choose random functions $\hat{f} : [N^2] \to [N^k]$ and $\hat{g} : [N^2] \times [N^2] \to [N^{k'}]$, as well as a random Boolean $s \in \{\mathsf{true}, \mathsf{false}\}$, which has the same purpose as in the proof of Theorem 2. Let $\mathrm{Im}(\hat{f})$ denote the image of function $\hat{f}$. We then supply $\mathcal{A}$ with a fake "conversation" between "Alice" and "Bob": we choose randomly $\sqrt{N}$ points $y'_1, y'_2, \ldots, y'_{\sqrt{N}}$ in $[N^k]$, $N - \sqrt{N}$ points $y_1, y_2, \ldots, y_{N-\sqrt{N}}$ in $\mathrm{Im}(\hat{f}) \subset [N^k]$, and one point $w \in [N^{k'}]$. We pretend that Alice has sent the list $Y = \{y_1, y_2, \ldots, y_{N-\sqrt{N}}\} \cup \{y'_1, y'_2, \ldots, y'_{\sqrt{N}}\}$ to Bob (in random order) and that Bob has responded with $Y' = \{y'_1, y'_2, \ldots, y'_{\sqrt{N}}\}$ (in increasing order) and w.

Now, we wait for $\mathcal{A}$'s queries to f and g.

- When $\mathcal{A}$ asks for $f(i)$ for some $i \in [N^2]$, there are two possibilities:
  - If $h'(i) = (0, 0)$, return $\hat{f}(i)$ to $\mathcal{A}$ as value for $f(i)$.
  - Otherwise, return $y'_{\pi_1(h'(i))}$.
- When $\mathcal{A}$ asks for $g(i, j)$ for some $i, j \in [N^2]$, there are two possibilities:
  - If $\pi_2(h'(i)) = \pi_2(h'(j)) \neq 0$ and either s is true and $i < j$ or s is false and $i > j$, return w as value for $g(i, j)$.
  - Otherwise, return $\hat{g}(i, j)$.

Suppose $\mathcal{A}$ happily returns the pair (i, j) for which it was told that $g(i, j) = w$, which is what a successful eavesdropper is supposed to do. This pair is in fact the answer to the search problem concerning function h'. Indeed, $g(i, j) = w$ for only the pair (i, j) for which $\pi_2(h'(i)) = \pi_2(h'(j)) \neq 0$, except with

the negligible probability that $\hat{g}(i',j') = w$ for some query (i',j') that $\mathcal{A}$ asks about g. However, we need an additional condition for the reduction to create an environment identical to the real one: if $y \in Y$ then $h'(f^{-1}(y)) = (0,0)$. This is required for all elements in $Y \setminus Y'$ to be accessible when $\mathcal{A}$ is querying f in the reduction. Fortunately, it is easy to see that this condition is satisfied except with vanishing probability when k is large enough.

Provided this condition is satisfied, queries asked by $\mathcal{A}$ concerning f and g are answered in the same way as they would be if both f and g were random functions consistent with the Y, Y' and w announced by Alice and Bob during the execution of the protocol. To see this, remember that Y and Y' (subsets of $[N^k]$) and w (element of $[N^{k'}]$) are uniformly picked at random in both the simulated and the real worlds. Moreover, the simulated function f is such that $f(i)$ is random when $h'(i) = (0,0)$. Among these $N^2 - \sqrt{N}$ input values, there are exactly $N - \sqrt{N}$ output values in $Y \setminus Y'$, as expected by $\mathcal{A}$. The remaining $\sqrt{N}$ input values i also satisfy $f(i) \in Y'$ as it should be. On the other hand, the simulated function g is random everywhere except for one single input pair $(i,j), i \neq j$, for which $g(i,j) = w$, as it is also expected by $\mathcal{A}$. Therefore, $\mathcal{A}$ will behave in the environment provided by the simulation exactly as in the real case. Since we disregard the negligible possibility that g might not be be one-to-one, the reduction solves the search problem concerning h' whenever $\mathcal{A}$ succeeds in finding the key. Notice again that each (new) question asked by $\mathcal{A}$ to either f or g translates to one or two questions actually asked to h'.

It follows that any successful cryptanalytic strategy that makes $o(N^{13/12})$ total queries to f and g would solve the search problem with only $o(N^{13/12})$ queries to function h', which is impossible by Lemma 4, except with vanishing probability. This establishes the $\Omega(N^{13/12})$ lower bound on the cryptanalytic difficulty of breaking our key exchange protocol, which matches the upper bound provided by the explicit attack discussed in Sect. 4.1. □

5 Conclusion, Conjectures and Open Questions

We presented an improved protocol for quantum key distribution over a classical channel and the first secure protocol for classical key distribution against a quantum adversary. Is it possible that they are optimal? We conjecture that they are not.

Indeed, we have discovered two sequences of protocols Q_ℓ and C_ℓ for $\ell \geqslant 2$ (which we shall describe in a subsequent paper) with the following properties. In protocol Q_ℓ, a classical Alice exchanges a key with a quantum Bob after $O(N)$ accesses to a random oracle in such a way that our most efficient quantum eavesdropping strategy requires the eavesdropper to access the same random oracle $\Theta\big(N^{1+\frac{\ell}{\ell+1}}\big)$ expected times. In protocol C_ℓ, purely classical Alice and Bob exchange a key after $O(N)$ accesses to a random oracle in such a way that our most efficient quantum eavesdropping strategy requires the eavesdropper to access the same random oracle $\Theta\big(N^{\frac{1}{2}+\frac{\ell}{\ell+1}}\big)$ expected times.

Our attacks proceed by quantum walks in Johnson graphs similar to those exploited in the proofs of Theorems 1 and 3 to obtain optimal attacks against our protocols 1 and 2. If they are the best possible against our new protocols as well, then key distribution protocols à la Merkle can be arbitrarily as secure in our quantum world as they were in the whimsical classical world known to Merkle in 1974: arbitrarily close to quadratic security can be restored. The obvious open question is to prove the optimality of our attacks. It would also be interesting to find a quantum protocol that exactly achieves quadratic security... or better! Indeed, even though it has been proven in the classical case that quadratic security is the best that can be achieved [3], there is no compelling evidence yet that such a limitation exists in the quantum world.

If our quantum attacks against the classical protocols are optimal, classical Alice and Bob can exchange a secret key against a quantum eavesdropper with as good a security (in the limit) as it was known to be possible for *quantum* Alice and Bob before this work. The main open question would be to break the $\Omega(N^{3/2})$ barrier or prove that this is not possible.

Even though our protocols Q_ℓ and C_ℓ require classical Alice to access the random black-box function only N times, she has to work for a *time* in $\Theta(N^\ell)$ to complete her share of the protocol. (This could be reduced to $\Theta(N^{\ell/2})$ for Q_ℓ if both Alice and Bob used quantum computing capabilities, but this remains nonlinear as soon as $\ell \geqslant 3$.) Could similar protocols exist in which Alice would be efficient even outside the required calls to the black-box function?

Finally, our lower bounds prove that it is not possible for the eavesdropper to learn Alice and Bob's key (x, x'), except with vanishing probability, unless she queries the black-box functions significantly more than the legitimate parties. However, we have not addressed the possibility for the eavesdropper to obtain efficiently *partial information* about the key. We leave this important issue for further research.

Acknowledgements

We are grateful to Troy Lee and Mohammad Mahmoody-Ghidary for insightful discussions. G. B. is also grateful to Ralph Merkle for his most inspiring Distinguished Lecture at CRYPTO '05, which sparked this entire line of work.

G. B. is supported in part by Canada's Natural Sciences and Engineering Research Council of Canada (NSERC), the Institut transdisciplinaire d'informatique quantique (INTRIQ), the Canada Research Chair program, the Canadian Institute for Advanced Research (CIFAR) and the Quantum *Works* Network. P. H. is supported in part by NSERC, CIFAR, Quantum *Works*, and the Canadian Network Centres of Excellence for Mathematics of Information Technology and Complex Systems (MITACS). S. L. is supported in part by the European Union 7th framework program QCS, ANR Défis QRAC and ANR Jeune chercheur CRYQ. L. S. is supported in part by NSERC, Quantum *Works*, Fundamental Research on Quantum Networks and Cryptography (FREQUENCY) and INTRIQ.

References

1. S. Aaronson and Y. Shi, "Quantum lower bounds for the collision and the element distinctness problems", *Journal of the ACM* **51**(4):595–605, 2004.
2. A. Ambainis, "Quantum walk algorithm for element distinctness", *SIAM Journal on Computing* **37**:210–239, 2007.
3. B. Barak and M. Mahmoody–Ghidary, "Merkle puzzles are optimal — An $O(n^2)$–query attack on any key exchange from a random oracle", *Advances in Cryptology – Proceedings of Crypto 2009*, Santa Barbara, California, pp. 374–390, 2009.
4. R. Beals, H. Buhrman, R. Cleve, M. Mosca and R. de Wolf, "Quantum lower bounds by polynomials", *Journal of the ACM* **48**(4):778–797, 2001.
5. C. H. Bennett, E. Bernstein, G. Brassard and U. V. Vazirani, "Strengths and weaknesses of quantum computing", *SIAM Journal on Computing* **26**(5):1510–1523, 1997.
6. M. Boyer, G. Brassard, P. Høyer and A. Tapp, "Tight bounds on quantum searching", *Fortschritte Der Physik* **46**:493–505, 1998.
7. G. Brassard, P. Høyer, M. Mosca and A. Tapp, "Quantum amplitude amplification and estimation", in *Quantum Computation and Quantum Information*, Samuel J. Lomonaco, Jr. (editor), *Contemporary Mathematics* **305**:53–74, AMS, 2002.
8. G. Brassard and L. Salvail, "Quantum Merkle puzzles", *Proceedings of Second International Conference on Quantum, Nano, and Micro Technologies (ICQNM08)*, Sainte Luce, Martinique, February 2008, pp. 76–79.
9. H. Buhrman, C. Dürr, M. Heiligman, P. Høyer, F. Magniez, M. Sántha and R. de Wolf, "Quantum algorithms for element distinctness", `http://arxiv.org/abs/quant-ph/0007016v2`, 2000.
10. W. Diffie and M. E. Hellman, "New directions in cryptography", *IEEE Transactions on Information Theory* **22**(6):644–654, 1976.
11. L. K. Grover, "Quantum mechanics helps in searching for a needle in a haystack", *Physical Review Letters*, **79**(2):325–328, 1997.
12. M. Heiligman, "Finding matches between two databases on a quantum computer", `http://arxiv.org/abs/quant-ph/0006136v1`, 2000.
13. P. Høyer, T. Lee and R. Špalek, "Negative weights make adversaries stronger", *Proceedings of 39th Annual Symposium on Theory of Computing (STOC)*, June 2007, pp. 526–535. The complete version can be found at `http://arxiv.org/abs/quant-ph/0611054v2`.
14. T. Lee, R. Mittal, B. W. Reichardt and R. Špalek, "An adversary for algorithms", `http://arxiv.org/abs/1011.3020v1`, 2010.
15. R. Merkle, "C.S. 244 Project Proposal", 1974. Facsimile available at `http://www.merkle.com/1974`.
16. R. Merkle, "Secure communications over insecure channels", *Communications of the ACM* **21**(4):294–299, 1978.
17. M. Sántha, "Quantum walk based search algorithms", *Proceedings of 5th Theory and Applications of Models of Computation (TAMC08)*, Xian, April 2008, LNCS 4978, pp. 31–46.

A Quantum Query Complexity

In our protocols, the work of the different parties is quantified by the number of queries made to black-box random functions, which can be modelled by a random oracle. In this Appendix, we review the main results from quantum query complexity that we used to prove our results and we sketch a new technical result that is needed for our lower-bound proofs.

Upper Bounds

Our attacks can be modelled as quantum walks on Johnson graphs. The graph $J(n, r)$ is an undirected graph in which each node contains some number r of distinct elements of $[n]$ and there is an edge between two nodes if and only if they differ by exactly two elements. Intuitively, we may think of "walking" from one node to an adjacent node by dropping one element and replacing it by another. The task is to find a specific k-subset of $[n]$. The nodes that contain this subset are called *marked*.

A random walk P on a Johnson graph can be quantized and the cost of the resulting quantum algorithm can be written as a function of S, U and C. These are the cost of setting up the quantum register in a state that corresponds to the stationary distribution, moving unitarily from one node to an adjacent node, and checking if a node is marked in order to flip its phase if it is, respectively.

Theorem 5. *[2, 17] Let M be either empty, or the set of vertices that contain a fixed subset of constant size $k \leqslant r$. Then there is a quantum algorithm that finds, with high probability, the k-subset if M is not empty at an expected cost in the order of*

$$\mathsf{S} + \frac{1}{\sqrt{\varepsilon}}\left(\frac{1}{\sqrt{\delta}}\mathsf{U} + \mathsf{C}\right),$$

where $\delta = n/r(n-r)$ is the eigenvalue gap of the symmetric walk on $J(n, r)$ and $\varepsilon = \Omega(\frac{r^k}{n^k})$ is the probability that a node is marked.

Lower Bounds

The central technical part of our lower bound consists in analysing the complexity of a function closely related to the hardness of breaking the key distribution protocols. This function is obtained by composing element distinctness and a variant of the search problem.

Consider two integer parameters α and β and three functions $c : [\alpha] \to [\alpha]$, $v : [\alpha] \to [\beta]$ and $h : [\alpha] \times [\beta] \to [\alpha]'$ so that there exists a single pair (i, j), $1 \leqslant i < j \leqslant \alpha$, for which $c(i) = c(j)$, which is called a *collision*, and

$$h(i, j) = \begin{cases} c(i) & \text{if } j = v(i) \\ 0 & \text{otherwise}. \end{cases}$$

The task is to find the unique nonzero collision in h, having only access to a black-box that computes h. This can be thought of as *searching* among β possibilities for the sole nonzero $h(i,\cdot)$ for each i and then finding two of those *elements*, among α possibilities, that are not *distinct*. Our main technical lemma, below, gives a lower bound on the number of queries to h that are required.

Lemma 5. *Finding a nonzero collision in h, structured as above, requires $\Omega(\alpha^{2/3}\beta^{1/2})$ quantum queries to h, except with vanishing probability.*

A complete proof of this lemma will appear separately but we now proceed to sketch it. For technical reasons, it is more convenient to prove this lower bound for the related decision problem: we are given a function h of the type above, but it is either based on a function c that has a single collision (as above) or on a one-to-one function c (in which case h is collision-free, except for value 0 in its image). The task is to decide which is the case. Obviously, any algorithm that can solve the search problem with probability of success at least $p > 0$ can be used to solve the decision problem with error bounded by $\frac{1}{2} - \frac{p}{2}$: run the search algorithm; if a collision is found (and verified), output "collision", otherwise output either "collision" or "no collision" with equal probability after flipping a fair coin. It follows that any lower bound on the bounded-error decision problem applies equally well to the search problem.

Again for technical reasons, we shall change the notation in order to adapt it to the normal usage in the field of quantum query complexity. For instance, function $c : [\alpha] \to [\alpha]$ will be represented by an element of $[\alpha]^\alpha$. This makes it possible to think of the decision version of element distinctness as a Boolean function $\mathsf{ED} : [\alpha]^\alpha \to \{0,1\}$, although it is a partial function since there is a *promise* on the valid inputs to ED: Given α integers $(z_1,\ldots,z_\alpha) \in [\alpha]^\alpha$, the promise is that either all the elements are distinct; or that all the elements are distinct except two, say $z_i \neq z_j$. The goal is to decide which of the two cases occurs by making as few queries as possible to the function that returns z_i on input i.

Ambainis' element distinctness quantum algorithm [2] runs in $O(\alpha^{2/3})$ queries to the input, and Aaronson and Shi proved that this is optimal [1]. Although the lower bound was proven using the polynomial method [4], a recent theorem of Ref. [14] shows that the generalized adversary bound is tight. Since our proof of the lower bound is derived using the generalized adversary method [13], we may conclude that there exists an adversary bound for element distinctness.

We compose the element distinctness problem with α instances of a promise version of a Search problem, which we call $\mathsf{pSEARCH}$. $\mathsf{pSEARCH} : [\alpha]'^\beta \to [\alpha]$ is also a promise problem. On input $(a_1,\ldots a_\beta)$, the promise is that all but one of the numbers are zero. The goal is to find and output this non-zero number by making queries that take i as input and return a_i.

The composed function we study is $\mathsf{H} = \mathsf{ED} \circ \mathsf{pSEARCH}^\alpha$. We now restate Lemma 5 in its decision-problem version.

Lemma 6. *The quantum query complexity of* H *is in* $\Omega(\alpha^{2/3}\beta^{1/2})$.

The proof uses the generalized adversary method for quantum query complexity, which we briefly describe here. Suppose we want to determine the quantum query complexity of a function F. We will assign weights to pairs of inputs in such a way as to bring out how hard it is (in terms of number of queries) to distinguish these inputs apart from one another. The adversary lower bound is the worst ratio of the spectral norm of this matrix, which measures the overall progress necessary in order for the algorithm to be correct, to the spectral norms of a associated matrices, which measure the maximum amount of progress that can be achieved by making a single query.

Definition 1. *Fix a function* $\mathsf{F} : S^n \to T$. *A symmetric matrix* $\Gamma : S^n \times S^n \to \mathbb{R}$ *is an adversary matrix for* F *provided* $\Gamma[x, y] = 0$ *whenever* $\mathsf{F}(x) = \mathsf{F}(y)$. *Let* $D_i[x, y] = 1$ *if* $x_i \neq y_i$ *and 0 otherwise. The adversary bound of* F *using* Γ *is*

$$\mathit{Adv}^{\pm}(\mathsf{F}; \Gamma) = \min_i \frac{\|\Gamma\|}{\|\Gamma \circ D_i\|},$$

where $\circ$ *denotes entrywise (or Hadamard) product, and* $\|A\|$ *denotes the spectral norm of* A *(which is equal to its largest eigenvalue). The adversary bound* $\mathit{Adv}^{\pm}(\mathsf{F})$ *is the maximum, over all adversary matrices* Γ *for* F, *of* $\mathit{Adv}^{\pm}(\mathsf{F}; \Gamma)$.

Since H is defined as the composition of ED and $\mathsf{pSEARCH}$, one would like to apply a composition theorem for the generalized adversary method [13], which would say that if a function $\mathsf{H} = \mathsf{F} \circ \mathsf{G}^{\alpha}$, then $\mathsf{Adv}^{\pm}(\mathsf{H}) = \mathsf{Adv}^{\pm}(\mathsf{F})\mathsf{Adv}^{\pm}(\mathsf{G})$ (up to constant factors, which will no longer be mentioned). Unfortunately, the composition theorem of Ref. [13] requires the inner and outer functions to be Boolean, which is not the case here for the inner function. (In fact, the outer function does not need to be Boolean according Corollary 5.6 in Ref. [14], but there are no general results known to the authors that yield a similar theorem when the inner function is not Boolean.)

Nevertheless, we are still able to prove the lower bound using techniques from Ref. [13] . Although our inner function is not Boolean, it has a lot of structure, which turns out to be sufficient for the proof to go through, modulo some modifications, which we briefly sketch here. (For the full version of the composition theorem, we refer the reader to the `ArXiv` version of Ref. [13].)

Our goal is to construct an adversary matrix Γ_{H} that captures the difficulty of applying ED to α instances of $\mathsf{pSEARCH}$. Recall that $\mathsf{Adv}^{\pm}$ is tight for query complexity, so we know that there exists an adversary matrix Γ_{ED} for which $\mathsf{Adv}^{\pm}(\mathsf{ED}; \Gamma_{\mathsf{ED}}) \geqslant \alpha^{2/3}$. We don't have an explicit expression for this matrix, let alone its spectral decomposition, but we know it exists.

For the inner $\mathsf{pSEARCH}$ problem, we construct an adversary matrix that we call Γ_{G} to keep consistent with the notation of Ref. [13]. We can analyse this matrix and give its explicit spectral decomposition. The block structure of the matrix and the form of its eigenvalues is key to proving the lower bound, so our proof does not hold for arbitrary non-Boolean inner functions g.

There are two main parts to the proof that $\mathsf{Adv}^{\pm}(\mathsf{H}) = \alpha^{2/3}\beta^{1/2}$. First we give a lower bound on $\|\Gamma_{\mathsf{H}}\|$, then we give an upper bound on $\|\Gamma_{\mathsf{H}} \circ D_i\|$, for each i.

Claim. $\|\Gamma_{\mathsf{H}}\| = \|\Gamma_{\mathsf{ED}}\|\|\Gamma_{\mathsf{G}}\|^{\alpha}$.

Proving this claim is the central and most technical part of the proof. In order to compute Γ_{H}'s spectral norm, we give its spectral decomposition. As in Ref. [13], we provide $(\alpha\beta)^{\alpha}$ eigenvectors and show that they form a basis. Our basis differs from that of Ref. [13], and is tailored to the properties we know about the spectral decomposition of Γ_{G}. We make essential use of the block structure of Γ_{G} and the fact that there are just two kinds of block: diagonal blocks, and off-diagonal blocks. (In the case of Boolean outputs, the same structure occurs, but there are just four blocks in that case, whereas we can handle many.)

Once we have the norm of Γ_{H}, we can use similar ideas to compute the value of $\|\Gamma_{\mathsf{H}} \circ D_i\|$. Because of the symmetry of H, it suffices to compute $\|\Gamma_{\mathsf{H}} \circ D_i\|$ for a fixed i. Fortunately, $\|\Gamma_{\mathsf{H}}\|$ and $\|\Gamma_{\mathsf{H}} \circ D_i\|$ share sufficient structure so that once the calculation of $\|\Gamma_{\mathsf{H}}\|$ is done, the calculation of $\|\Gamma_{\mathsf{H}} \circ D_i\|$ follows easily.

For any query i, we decompose it into the index p in which it occurs within x, and the index of the position queried within x_p. Then, D_i decomposes naturally into two parts, D_p and D_q.

Claim. $\forall i$ that decomposes into p, q, $\|\Gamma_{\mathsf{H}} \circ D_i\| = \|\Gamma_{\mathsf{ED}} \circ D_p\|\|\Gamma_{\mathsf{G}} \circ D_p\|\|\Gamma_{\mathsf{G}}\|^{\alpha-1}$.

Combining the two claims, we get

$$\mathsf{Adv}^{\pm}(\mathsf{H}; \Gamma_{\mathsf{H}}) = \mathsf{Adv}^{\pm}(\mathsf{ED})\mathsf{Adv}^{\pm}(\mathsf{pSEARCH}) = \mathsf{Q}(\mathsf{ED})\mathsf{Q}(\mathsf{OR}),$$

where Q denotes the quantum query complexity, and where the final inequality follows from the fact that OR is a special case of $\mathsf{pSEARCH}$. The lemma follows by using the known quantum query complexity lower bounds for $\mathsf{Q}(\mathsf{OR})$, which is in $\Omega(\beta^{1/2})$ [5], and for $\mathsf{Q}(\mathsf{ED})$, which is in $\Omega(\alpha^{2/3})$ [1].