THEORY OF COMPUTING, Volume 7 (2011), pp. 19–25
www.theoryofcomputing.org

NOTE

Inverting a Permutation is as Hard as Unordered Search

Ashwin Nayak*

Received: July 28, 2010; published: March 10, 2011.

Abstract: We show how an algorithm for the problem of inverting a permutation may be used to design one for the problem of unordered search (with a unique solution). Since there is a straightforward reduction in the reverse direction, the problems are essentially equivalent.

The reduction we present helps us bypass the hybrid argument due to Bennett, Bernstein, Brassard, and Vazirani (1997) and the quantum adversary method due to Ambainis (2002) that were earlier used to derive lower bounds on the quantum query complexity of the problem of inverting permutations. It directly implies that the quantum query complexity of the problem is asymptotically the same as that for unordered search, namely in $\Theta(\sqrt{n})$.

*Research supported in part by NSERC Canada, CIFAR, an ERA (Ontario), QuantumWorks, MITACS, and ARO (USA). Research at Perimeter Institute for Theoretical Physics is supported in part by the Government of Canada through Industry Canada and by the Province of Ontario through MRI.

ACM Classification: F.2.2, F.1.3, F.1.2

AMS Classification: 68Q12, 68Q17, 81P68, 68Q15, 68Q25

Key words and phrases: unordered search, inverting permutations, query complexity, quantum algorithm, hybrid argument, randomized reduction

 DOI: 10.4086/toc.2011.v007a002

1 Introduction

Let n be a positive integer. The problem PERMUTATION$_n$ of inverting a permutation π on the set $[n] = \{1,2,\ldots,n\}$ is defined as follows. Given π in the form of an oracle, and n as input, output "yes" if the pre-image $\pi^{-1}(1)$ is even and "no" if it is odd. This is a natural decision version of the problem that asks us to *find* $\pi^{-1}(1)$. A related problem is that of unordered search: Given a function $f : [n] \to \{0,1\}$ as an oracle, and n as input, output "yes" if $f^{-1}(1)$ is non-empty and "no" otherwise. In other words, determine if f maps any element $i \in [n]$ to 1. In this article, we restrict ourselves to functions f which map at most one element to 1. As we might expect, these constitute the hardest instances of unordered search. We refer to the corresponding sub-problem as UNIQUE SEARCH$_n$.

The two problems were originally used by Bennett, Brassard, Bernstein, and Vazirani [2] to show limitations of quantum computers. The search problem UNIQUE SEARCH was used to show that relative to a random boolean oracle A, with probability 1, $\mathbf{NP}^A \not\subseteq \mathbf{BQP}^A$. The inversion problem PERMUTATION was similarly used to show that relative to a random permutation oracle A, with probability 1, $\mathbf{NP}^A \cap \mathbf{co\text{-}NP}^A \not\subseteq \mathbf{BQP}^A$. For the first result, Bennett *et al.* showed that any quantum algorithm for UNIQUE SEARCH$_n$ requires $\Omega(\sqrt{n})$ queries for a constant probability of error. This involved a hybrid argument that works for both worst-case error and distributional error under an equal mixture of uniform distributions over "yes" and "no" instances. The lower bound is matched by the Grover quantum search algorithm [3], and is therefore optimal. For the second result, Bennett *et al.* used a nested hybrid argument and showed that any quantum algorithm for the inversion problem requires $\Omega(\sqrt[3]{n})$ queries (for constant probability of error under the uniform distribution). The optimal bound of $\Omega(\sqrt{n})$ was established for worst-case query complexity by Ambainis [1] using the then-newly-minted quantum adversary method.

An algorithm for unordered search (in fact, for UNIQUE SEARCH$_n$) may be used to solve the inversion problem PERMUTATION$_n$ in the obvious manner, using at most twice the number of oracle queries. Namely, we define a boolean function f on $[n]$ such that $f(i) = 1$ iff $\pi(i) = 1$ and i is even. This function may be evaluated with one classical query to an oracle for π. An additional query is used in the quantum case to "erase" the answer to the first query (see the discussion regarding quantum queries at the end of the proof of Theorem 2.1 in Section 2). Therefore the Grover quantum search algorithm [3] solves this problem with $O(\sqrt{n})$ queries to an oracle for π. We describe a reduction in the reverse direction, i. e., we show how any algorithm that solves PERMUTATION$_n$ may be modified to solve UNIQUE SEARCH$_{n/2}$ when n is even. The intuition behind the reduction comes from a hybrid argument in which we consider runs of the algorithm on oracles not in its domain. This kind of device has been used in numerous works on quantum query complexity to great effect.

The reduction we present is randomized and is between distributional versions of the problems, with equal weight on "yes" and "no" instances, and with uniform conditional distributions for each kind of instance. Due to the inherent symmetry in the two problems under consideration, the distributional and worst-case versions of the problems are, in fact, equivalent in query complexity (for the distributions described above; see Lemma 2.2 and the discussion following it). Thus, we are also able to derive an algorithm for UNIQUE SEARCH$_{n/2}$ with a bound on its worst-case error.

Let μ_n denote the distribution obtained by taking an equal mixture of the sole "no" instance and the uniform distribution over "yes" instances of UNIQUE SEARCH$_n$.

Theorem 1.1. *Let n be an even positive integer. Let $\mathcal{A}$ be an algorithm (classical or quantum) that solves* PERMUTATION$_n$ *with distributional error at most $\varepsilon < 1/2$ on the uniform distribution over permutations on $[n]$, with q queries to the permutation oracle. Then there is an algorithm of the same kind as $\mathcal{A}$ (classical or quantum) with distributional error at most $(1+2\varepsilon)/4$ with respect to the distribution $\mu_{n/2}$, that solves* UNIQUE SEARCH$_{n/2}$ *with at most q queries to the search oracle in the classical case, and at most $2q$ queries in the quantum case. Moreover, this may be modified to an algorithm for* UNIQUE SEARCH$_{n/2}$ *with worst-case error at most $1/(3-2\varepsilon) < 1/2$ and with the same query complexity.*

We emphasize that the reduction uses only the knowledge of a bound ε on the distributional error of $\mathcal{A}$, and not its precise value. An algorithm for PERMUTATION$_n$ for $n > 2$ with distributional error at most ε on the uniform distribution may be used to solve PERMUTATION$_{n-1}$ with the same query complexity and distributional error at most $\varepsilon + 1/2n$ on the uniform distribution (see the discussion after Lemma 2.2). So a reduction similar to the one in the theorem above exists for odd n as well.

Any quantum algorithm for UNIQUE SEARCH$_{n/2}$ with constant probability of error $< 1/2$ with respect to $\mu_{n/2}$ requires $\Omega(\sqrt{n})$ queries [2]. Theorem 1.1 therefore implies a similar lower bound for quantum algorithms for PERMUTATION$_n$, and an $\Omega(n)$ query lower bound for classical algorithms for the problem. (The lower bound of $\Omega(n)$ for PERMUTATION$_n$ in the classical case is straightforward to prove directly.)

Corollary 1.2. *Any quantum algorithm that solves* PERMUTATION$_n$ *(for any integer $n > 1$) with constant distributional error $\varepsilon < 1/2$ on the uniform distribution over permutations on $[n]$ requires $\Omega(\sqrt{n})$ queries. Consequently, the same query lower bound holds for algorithms for* PERMUTATION$_n$ *with worst-case error at most ε.*

Naturally, Theorem 1.1 and Corollary 1.2 both also hold for any quantum algorithm for the *search* version of PERMUTATION$_n$ with the same kind of error bound as in the statements above.

The reduction we present bypasses the hybrid argument due to Bennett, Bernstein, Brassard, and Vazirani [2] and the quantum adversary method due to Ambainis [1], and shows a direct connection between inversion and search. The hybrid argument underlying the reduction was discovered in 2004 and communicated informally to a few people. The reduction is written up here for wider dissemination.

2 The reduction

We start by fixing some notation and making preliminary observations. Then we sketch a hybrid argument which paves the way for the reduction (Theorem 2.1). Lemma 2.2 derives a worst-case algorithm for UNIQUE SEARCH$_n$ from an average-case algorithm, and together with Theorem 2.1 implies Theorem 1.1. We finish by sketching how Theorem 1.1 extends to odd n.

Let n be any positive integer. Define the following sets of "no" and "yes" instances of PERMUTATION$_n$:

$$\begin{aligned} P_0 &= \left\{ \pi : \pi \text{ is a permutation on } [n], \quad \pi^{-1}(1) \text{ is odd} \right\}, \quad \text{and} \\ P_1 &= \left\{ \pi : \pi \text{ is a permutation on } [n], \quad \pi^{-1}(1) \text{ is even} \right\}. \end{aligned}$$

We also consider oracles that compute functions $h : [n] \to [n]$ with a unique collision at 1, with one odd and one even number in the colliding pair. In other words, these functions h are such that there are precisely two distinct elements i, j with the same image under h. Moreover, $h(i) = h(j) = 1$, and precisely one of i, j is odd (and the other is even). Let Q denote the set of all such functions.

Consider any fixed permutation π on $[n]$. Consider also the functions in Q that differ from π in exactly one point. These are functions h with a unique collision such that the collision is at 1, and $\left|\pi^{-1}(1) \cap h^{-1}(1)\right| = 1$. If $\pi \in P_0$, then the even element that is also mapped to 1 by h is precisely the one on which π and h differ. Similarly, if $\pi \in P_1$, then the odd element that is also mapped to 1 by h is precisely the one on which π and h differ. Let Q_π denote the set of such functions h. If we pick a uniformly random permutation $\pi \in P_0$, and then pick a uniformly random function h in Q_π, then h is uniformly random in Q. The same holds if we switch P_0 with P_1.

Given a permutation π on $[n]$ with n even, and a function $f : [n/2] \to \{0,1\}$, we define a function $h_{\pi,f} : [n] \to [n]$ as follows. If $\pi \in P_0$, for any $i \in [n]$,

$$h_{\pi,f}(i) \quad = \quad \begin{cases} 1 & \text{if } i \text{ is even and } f(i/2) = 1, \\ \pi(i) & \text{if } (i \text{ is odd}) \text{ or } (i \text{ is even and } f(i/2) = 0). \end{cases} \tag{1}$$

If $\pi \in P_1$, for any $i \in [n]$,

$$h_{\pi,f}(i) \quad = \quad \begin{cases} 1 & \text{if } (i \text{ is odd}) \text{ and } f((i+1)/2) = 1, \\ \pi(i) & \text{if } (i \text{ is even}) \text{ or } (i \text{ is odd and } f((i+1)/2) = 0). \end{cases} \tag{2}$$

The function $h_{\pi,f}$ coincides with π if $f^{-1}(1)$ is empty and belongs to Q_π otherwise.

We make use of the above properties of P_0, P_1, Q, Q_π and $h_{\pi,f}$ in our reduction.

The essential idea behind our reduction is that any algorithm that distinguishes a uniformly random permutation in P_0 from a uniformly random one in P_1 necessarily distinguishes a random permutation from P_i from a uniformly random unique-collision function in Q, for at least one $i \in \{0,1\}$. Suppose that $i = 0$. Using convexity, we may further deduce that such an algorithm also distinguishes the permutation $\pi \in P_0$ from a uniformly random unique-collision function $h \in Q_\pi$, for at least one π. Since any function $h \in Q_\pi$ differs from π in exactly one of $n/2$ points when n is even, this final problem is equivalent to the problem of unique unordered search over a domain of size $n/2$. The above hybrid argument suffices to prove a query lower bound for PERMUTATION$_n$, but is not entirely satisfactory because it corresponds to a non-uniform reduction. (*A priori*, we do not know which $i \in \{0,1\}$ and which $\pi \in P_i$ would give us a correct algorithm for UNIQUE SEARCH$_{n/2}$.) We may however glean a reduction in the uniform sense from this argument. The idea is to replace the choices made on the basis of existential arguments by randomized ones. We therefore try distinguishing a uniformly random $\pi \in P_0$ from a uniformly random $h_{\pi,f} \in Q_\pi$, and similarly with a uniformly random permutation from P_1, with equal probability. At least one of these probabilistic attempts succeeds, and gives us a bounded-error algorithm.

Recall that μ_n is the distribution which assigns probability $1/2$ to the constant function 0, and probability $1/2n$ to each of the "yes" instances of UNIQUE SEARCH$_n$. Let μ_n^1 be the uniform distribution on "yes" instances alone, and let μ_n^0 be the analogue for the lone "no" instance.

Theorem 2.1. *Let n be an even positive integer. Let $\mathcal{A}$ be an algorithm (classical or quantum) that solves* PERMUTATION$_n$ *with distributional error at most $\varepsilon < 1/2$ on the uniform distribution over permutations on $[n]$, with q queries to the permutation oracle. Then there is an algorithm of the same kind as $\mathcal{A}$ with distributional error at most $(1+2\varepsilon)/4 < 1/2$ with respect to $\mu_{n/2}$ that solves* UNIQUE SEARCH$_{n/2}$ *with at most q queries to the search oracle in the classical case, and at most $2q$ queries in the quantum case. Moreover, its error probability on an input drawn from $\mu^0_{n/2}$ is at most ε and that for $\mu^1_{n/2}$ is $1/2$.*

Proof. Suppose we are given an algorithm $\mathcal{A}$ as in the statement of the theorem that takes as input an even integer $n \geq 1$, and an oracle $g : [n] \to [n]$. Let f be an input oracle for UNIQUE SEARCH$_{n/2}$. Recall the definition of the function $h_{\pi,f} : [n] \to [n]$, where π is a permutation on $[n]$, as in Eqs. (1) and (2). The following reduction $\mathcal{B}$ solves UNIQUE SEARCH$_{n/2}$ with distributional error as claimed:

With probability $\frac{1}{2}$, pick a uniformly random permutation $\pi \in P_0$, output $\mathcal{A}(n, h_{\pi,f})$ and stop.

With probability $\frac{1}{2}$, pick a uniformly random permutation $\pi \in P_1$, output $\neg\mathcal{A}(n, h_{\pi,f})$ and stop.

We may calculate the probability of error of the algorithm $\mathcal{B}$ by considering "yes" and "no" instances separately. The output of $\mathcal{B}$ on the lone "no" instance of UNIQUE SEARCH$_{n/2}$ is

$$\begin{cases} \mathcal{A}(n,\pi) & \text{with probability } \frac{1}{2}\text{, for a uniformly random } \pi \in P_0\text{, and} \\ \neg\mathcal{A}(n,\pi) & \text{with probability } \frac{1}{2}\text{, for a uniformly random } \pi \in P_1. \end{cases}$$

Let ε_0 be the probability of error of $\mathcal{A}$ on a uniformly random oracle from P_0, and let ε_1 be the corresponding quantity for P_1. We have $\varepsilon_0 + \varepsilon_1 \leq 2\varepsilon$. The probability of error of $\mathcal{B}$ on the "no" instance is thus bounded by $(\varepsilon_0 + \varepsilon_1)/2 \leq \varepsilon$.

The output of $\mathcal{B}$ on a uniformly random "yes" instance of UNIQUE SEARCH$_{n/2}$ is

$$\begin{cases} \mathcal{A}(n,h) & \text{with probability } \frac{1}{2}\text{, for a uniformly random } h \in Q\text{, and} \\ \neg\mathcal{A}(n,h) & \text{with probability } \frac{1}{2}\text{, for a uniformly random } h \in Q. \end{cases}$$

If p denotes the probability that the output $\mathcal{A}(n,h)$ is "no" for a uniformly random $h \in Q$, the probability of error of $\mathcal{B}$ on a uniformly random "yes" instance of UNIQUE SEARCH$_{n/2}$ is $p/2 + (1-p)/2 = 1/2$.

Since π is known explicitly in the algorithm $\mathcal{B}$, the function $h_{\pi,f}$ can be evaluated with at most one query to f in the classical case. In the quantum case, the situation is not as straightforward. Data in quantum memory generated during computations may prevent the quantum interference necessary for correct working of an algorithm. This may be the case with the answers to oracle queries to f that are used to compute $h_{\pi,f}$. It is therefore important to simulate queries to $h_{\pi,f}$ cleanly, i. e., without modifying any part of the workspace except the answer register. With a standard implementation of the oracle to f as a unitary operator, a query to $h_{\pi,f}$ can be simulated cleanly by using *two* queries (see [2, Section 4] or [4, Section 1.5]). We note that by working with a different implementation of the oracle for f, it is possible to simulate a query to $h_{\pi,f}$ cleanly with only one query. As this leads to only a minor improvement in our reduction, we omit the details. $\square$

Any algorithm with worst-case error ε implies an algorithm with distributional error ε with respect to any distribution. We point out that UNIQUE SEARCH and PERMUTATION both admit random self-reductions, so the average-case algorithm described in Theorem 2.1 implies the worst-case algorithm claimed in Theorem 1.1.

Lemma 2.2. *Suppose $\mathcal{B}$ is an algorithm for* UNIQUE SEARCH$_n$ *with distributional error at most ε_1 on distribution μ_n^1, and at most ε_0 on μ_n^0 such that $\varepsilon_0 + \varepsilon_1 < 1$. Then, there is an algorithm for* UNIQUE SEARCH$_n$ *that makes the same number of queries as $\mathcal{B}$, and has worst-case error at most*

$$\varepsilon \quad = \quad \frac{\max\{\varepsilon_0, \varepsilon_1\}}{1 + |\varepsilon_0 - \varepsilon_1|} \quad < \quad \frac{1}{2}\,.$$

Proof. Composing a function $f : [n] \to \{0,1\}$ with a permutation σ on $[n]$ preserves the "yes" and the "no" instances of UNIQUE SEARCH$_n$. Consider the algorithm $\mathcal{B}_{\mathrm{sym}}$:

> Pick a uniformly random permutation σ on $[n]$, and then return the value given by the algorithm $\mathcal{B}$ with every oracle query $i \in [n]$ replaced by a query to $\sigma(i)$.

Effectively, any single instance of UNIQUE SEARCH$_n$ is mapped to a uniformly random instance with the same answer. So the worst-case error on the "no" instance is at most ε_0, and on any "yes" instance is at most ε_1.

It only remains to equalize the bounds on error on the two kinds of instance, and this may be accomplished by a standard modification: depending on whether $\varepsilon_0 < \varepsilon_1$ or not, we accept or reject with some probability p, and run the algorithm $\mathcal{B}_{\mathrm{sym}}$ with probability $1 - p$. The choice of

$$p = \frac{|\varepsilon_1 - \varepsilon_0|}{1 + |\varepsilon_1 - \varepsilon_0|}$$

gives us the claimed error bound. □

There is a similar randomized self-reduction for PERMUTATION$_n$. Let ω, σ be uniformly random permutations on $[n]$ such that ω maps 1 to itself, and σ permutes odd integers among themselves and even integers among themselves. Then for any permutation π on $[n]$, the composition $\omega \circ \pi \circ \sigma$ is uniformly distributed in P_1 if π is a "yes" instance, and in P_0 if it is a "no" instance. Therefore, an algorithm $\mathcal{A}$ for PERMUTATION$_n$ with distributional error at most $\varepsilon_1, \varepsilon_0$ on uniformly random "yes" and "no" instances, respectively, leads to a worst-case algorithm which makes error at most ε_1 on any "yes" instance and at most ε_0 on any "no" instance. When n is even, the distributional error of $\mathcal{A}$ on the uniform distribution is at most $\varepsilon = (\varepsilon_0 + \varepsilon_1)/2$ and when n is odd, it is at most

$$\varepsilon = \frac{(1 + 1/n)\varepsilon_0 + (1 - 1/n)\varepsilon_1}{2}\,.$$

We may now rebalance the bound on error in the two kinds of instance to obtain a worst-case algorithm. The self-reduction also allows us to use an algorithm $\mathcal{A}$ for PERMUTATION$_n$ to solve PERMUTATION$_{n-1}$ as claimed in Section 1. We first symmetrize the algorithm $\mathcal{A}$ through the self-reduction, and then run it on the extension of the input permutation π on $[n-1]$ obtained by defining $\pi(n) = n$.

Acknowledgements

I thank the anonymous referees for their insightful comments and suggestions. They vastly improved the quality of the presentation.

References

[1] ANDRIS AMBAINIS: Quantum lower bounds by quantum arguments. *J. Comput. System Sci.*, 64(4):750–767, 2002. [doi:10.1006/jcss.2002.1826] 20, 21

[2] CHARLES H. BENNETT, ETHAN BERNSTEIN, GILLES BRASSARD, AND UMESH VAZIRANI: Strengths and weaknesses of quantum computing. *SIAM J. Comput.*, 26(5):1510–1523, 1997. [doi:10.1137/S0097539796300933] 20, 21, 23

[3] LOV K. GROVER: A fast quantum mechanical algorithm for database search. In *Proc. 28th STOC*, pp. 212–219. ACM Press, 1996. [doi:10.1145/237814.237866] 20

[4] PHILLIP KAYE, RAYMOND LAFLAMME, AND MICHELE MOSCA: *An Introduction to Quantum Computing*. Oxford University Press, New York, NY, USA, 2007. 23

AUTHOR

Ashwin Nayak
Associate Professor
Department of Combinatorics and Optimization, and Institute for Quantum Computing,
University of Waterloo, Waterloo, ON, Canada
Perimeter Institute for Theoretical Physics, Waterloo, ON, Canada
ashwin.nayak@uwaterloo.ca
http://www.math.uwaterloo.ca/~anayak

ABOUT THE AUTHOR

ASHWIN NAYAK received his B. Tech. in 1995 from Indian Institute of Technology, Kanpur, and his Ph. D. in 1999 from University of California, Berkeley. His Ph. D. dissertation on quantum computation and communication was supervised by Umesh V. Vazirani, and he continues to research quantum algorithms and protocols. When he can wrest himself away from his computer, he dabbles in yoga. He hopes that its teaching of detachment will one day free him of this device.