Computing the order of a solvable black-box group in quantum polynomial time, and applications

Brad Hannigan-Daley (20177981)

April 26, 2009

Abstract

We explore a quantum polynomial-time algorithm by Watrous for computing the order of a solvable black-box group, and discuss applications to related group-theoretic problems.

1 Introduction

Given a finite solvable black-box group, Watrous [1] describes an algorithm for determining the order of the group in quantum polynomial time, with a modification of Shor's order-finding algorithm at its core. As a byproduct, the algorithm also produces a uniform superposition of the elements of the group. About seventeen years prior to the publication of Watrous's paper, it had been proven by Babai and Szemerédi that the problem of computing the order of a solvable group, in a classical setting, is of complexity class **NP** even in the more specific case that the group is abelian [2]. There is no known classical polynomial time algorithm for solving this problem, and hence this provides further evidence of the computational advantage of quantum computers over classical computers. It was also known that the problem solved here is *low* for the complexity class **PP** of decision problems that are solvable to an arbitrary degree of accuracy by a probabilistic Turing machine in polynomial time [3]. That is to say, gaining the ability to instantaneously compute the order of a finite solvable black-box group does not provide any additional advantage to an algorithm which solves a problem in **PP**.

The problem of determining the order of a (solvable) group is an important one not only because it is interesting in and of itself, but many other group-theoretic problems can be reduced to it. For example, given a list of elements $g_1, \ldots, g_k, h$ in some finite black-box group, the problem of testing whether h lies in the subgroup generated by $g_1, \ldots, g_k$ reduces to the problem of computing and comparing the orders of two subgroups. Other such problems will be discussed.

This report provides an analysis of Watrous's algorithm, and will explain certain details of Watrous's paper that were assumed to be known or obvious to the reader. As the problem at hand is of a group-theoretic nature, we begin by providing some definitions and elementary results of group theory for the reader who does not have a background in this area, along with an explanation of the "black-box group" framework in which this algorithm operates.

2 Preliminaries

A *group* is a set G equipped with a binary operation, which we denote here by $\cdot$, satisfying the following axioms:

- (Closure) For all $g, h \in G$, $g \cdot h \in G$.
- (Existence of identity) There exists $1 \in G$ such that $g \cdot 1 = 1 \cdot g = g$ for all $g \in G$.
- (Existence of inverses) For each $g \in G$ there exists $g' \in G$ such that $g \cdot g' = g' \cdot g = 1$.
- (Associativity) For all $f, g, h \in G$ we have $f \cdot (g \cdot h) = (f \cdot g) \cdot h$.

It can be proven [4] that the element 1 from the first axiom is uniquely determined, and it is known as the *identity* element of G. It can also be shown that for each g the element g' from the second axiom is uniquely determined; it is known as the *inverse* of g and is denoted by g^{-1}. If the operation $\cdot$ is understood from the context, then to simplify notation we may simply write gh instead of $g \cdot h$. For example, the fourth axiom can then be written "$f(gh) = (fg)h$ for all $f, g, h \in G$." Using this notation, it is conventional to denote by g^n the product $gg \cdots g$ of an element g with itself n times, and to write $g^0 := 1$ and $g^{-n} := (g^{-1})^n$. The usual exponent laws then hold.

As an example of a group, consider the set S_n of bijections (invertible functions) from $\{1, 2, \ldots, n\}$ to $\{1, 2, \ldots, n\}$. By taking our binary operation to be composition of functions ($\circ$), S_n has the structure of a group: the identity function, which fixes each element, is the identity element of the group, the existence of inverses is guaranteed by definition, and composition of functions is associative. The group S_n is known as the *symmetric group* on $\{1, 2 \ldots, n\}$.

It is important to note that we do not impose the condition that $gh = hg$ for all $g, h \in G$. For example, the group S_n described above does not satisfy this, assuming $n \geq 3$: if f is the bijection with $f(1) = 2, f(2) = 1$,and $f(k) = k$ for all other k, and g is the bijection with $g(1) = 3, g(3) = 1$, and $g(k) = k$ for all other k, then we have $(f \circ g)(1) = 3$ and $(g \circ f)(1) = 2$, hence $f \circ g \neq g \circ f$. If a group does satisfy this additional constraint that $gh = hg$ for all elements g and h, then it is known as an *abelian* group. For example, the set $\mathbb{Z}$, equipped with the binary operation $+$ of addition, forms an abelian group with identity 0. By analogy to this, the operation in an abelian group is often denoted by $+$, with the inverse of g denoted by $-g$ and the identity denoted by 0 rather than 1.

A *subgroup* of G is a subset H of G which itself satisfies the above axioms, using the same binary operation. Note that the identity element of H is necessarily that of G, as it is unique in G. As an example, in the previous example of $\mathbb{Z}$ under $+$, the set $2\mathbb{Z}$ of even integers is a subgroup. The *order* of a group G is its cardinality $|G|$ as a set, *i.e.* the number of its elements. The order may be infinite, as in the case of $\mathbb{Z}$, but the body of this paper will deal only with finite groups. Given a subset S of a group G, the *subgroup generated by* S is the smallest subgroup of G containing S, which we denote by $\langle S\rangle$. For $S = \{g_1, \ldots, g_k\}$, we write $\langle S\rangle = \langle g_1, \ldots, g_k\rangle$. A group of the form $\langle g\rangle$ is called *cyclic*, and consists precisely of all elements of the form g^m for integers m. For example, the subgroup $2\mathbb{Z}$ of $\mathbb{Z}$ is cyclic, generated by 2. (As we usually use additive notation for abelian groups, we write mg instead of g^m, so in $\mathbb{Z}$ we have $\langle 2\rangle = \{2n : n \in \mathbb{Z}\}$.) For each $g \in G$, the *order* of g in G is the order of the subgroup $\langle g\rangle$, and is equal to the least positive integer m such that $g^m = 1$. An elementary result of group theory, known as Lagrange's theorem, states that if H is a subgroup of a finite group G, then $|H|$ is a divisor of $|G|$. It follows that the order of each element of G divides $|G|$.

Let H be a subgroup of G. A *left coset* of H in G is a set of the form $gH := \{gh : h \in H\}$ for some $g \in G$. The *right cosets* Hg are defined similarly. Note that the left cosets gH are not, in general, in one-to-one correspondence with the elements g; for example, $hH = H$ for each $h \in H$. It can be shown that the left (or right) cosets of H compose a partition of G into $|G|/|H|$ parts of size $|H|$. (In fact, this is generally how Lagrange's theorem, above, is proven.) If for each $g \in G$ the set $g^{-1}Hg := \{g^{-1}hg : h \in H\}$ is equal to H, we say that H is a *normal subgroup* of G and write $H \triangleleft G$. An equivalent way to phrase this is that $gH = Hg$ for all $g \in G$. Note that if N and H are subgroups of G such that $N \subset H$ and $N \triangleleft G$, it is immediate from the definition that $N \triangleleft H$. Now, suppose H is a subgroup of G, so it has $|G|/|H|$ cosets gH. We would like to turn the set of cosets of H into a group, using the obvious choice of multiplication given by $(g_1H)(g_2H) = g_1g_2H$. However, this is not always well-defined. It can be shown [4] that this operation is well-defined, and does indeed turn the set of cosets of H into a group, if and only if $H \triangleleft G$. This group is called the *factor group* or *quotient group* G/H. Observe that the coset $1H = H$ serves as the identity element of G/H.

For $g, h \in G$, the *commutator* of g and h is defined by $[g, h] = g^{-1}h^{-1}gh$. We define the *commutator subgroup* or *derived subgroup* of G to be the subgroup G' generated by all commutators in G. To motivate this definition, observe that for $g, h \in G$, we have $[g, h] = 1$ if and only if $gh = hg$. It follows that $G' = \{1\}$ if and only if all elements of G commute with each other, *i.e.* G is abelian, so in a heuristic sense G' provides a measure of "how far away G is from being abelian." It can easily be shown [4] that $G' \triangleleft G$, and that G/G' is necessarily abelian. (In fact, this latter factor group is known as the *abelianization* of G.)

Define $G^{(0)} := G$, and inductively define $G^{(n)} := (G^{(n-1)})'$ for all positive integers n. Then we have a *normal series*

$$G = G^{(0)} \triangleright G^{(1)} \triangleright G^{(2)} \triangleright \ldots$$

of subgroups of G. We say that G is *solvable* if $G^{(N)} = \{1\}$ for some positive integer N. [1] Clearly each abelian group G is solvable, since in this case we have $G^{(1)} = G' = \{1\}$, and so the concept of a solvable group generalizes, in a sense, the concept of an abelian

[1] The concept of solvable groups originally arose in Galois theory, which was developed to the end of answering questions about the solvability of certain polynomial equations. Each such polynomial equation can be naturally assigned a finite group called its *Galois group*, and the equation is solvable by radicals if and only if this group is solvable — hence the terminology.

group. It can be proven that if G is solvable, then so is every subgroup and every factor group of G.

Another characterization of solvability is as follows [4]: A finite group G is solvable if and only if there exist elements $g_1, \ldots, g_m \in G$ such that, putting $H_j = \langle g_1, \ldots, g_j \rangle$, we have $\{1\} =: H_0 \lhd H_1 \lhd \ldots \lhd H_m = G$. Observe that each factor group H_{j+1}/H_j is generated by the coset containing g_{j+1} and is therefore cyclic.

To apply quantum-computational techniques to group theory, we shall employ the formalism of a *black-box group*; this concept was first introduced by Babai and Szemerédi [2] in 1984. Given a finite group G, we encode its elements as distinct binary strings of some fixed length n, called the *encoding length*. Note that we necessarily have $2^n \geq |G|$, as there are only 2^n binary strings of length n. Furthermore, we have a black-box, or *group oracle*, which takes as input two binary strings corresponding to elements g and h of G, and returns the binary string which represents gh at unit cost. This is the framework as originally introduced in the context of classical circuits. We can use this framework in the context of quantum circuits by considering each element g of G as a pure n-qubit state $|g\rangle$ via this encoding into binary strings, and modeling the group oracle by the $2n$-qubit gate $U_G : |g\rangle |h\rangle \mapsto |g\rangle |gh\rangle$ extended linearly. Note that U_G, as given, is actually only defined on the basis elements $|g\rangle |h\rangle$ where g and h are legitimate encodings of elements of the group. However, as this gate will in practice only be used on states which are linear combinations of these basis elements, we can define U_G so as to have some arbitrary, but predefined, behaviour on other basis elements. For example, it could simply act as the identity on each of those states. Defined this way, U_G is indeed a unitary gate, and its inverse is given by $U_G^{-1} : |g\rangle |h\rangle \mapsto |g\rangle |g^{-1}h\rangle$ (with behaviour defined on "non-group" basis elements so as to be consistent with U_G). Lastly, for any subset S of G, we define the state

$$|S\rangle := \frac{1}{\sqrt{|S|}} \sum_{s \in S} |s\rangle ,$$

a normalized uniform superposition of the elements of S.

3 The algorithm

To begin Watrous's algorithm for computing the order of a finite solvable group G, we need a particular list $g_1, \ldots, g_m$ of generators for G, which we obtain as follows. Suppose we are given a group oracle for a finite black-box group G with encoding length n, along with a set of generators for G. There is a polynomial-time classical algorithm [5] which will construct, with high probability, elements $g_1^{(j)}, g_2^{(j)}, \ldots g_k^{(j)}$ of G for $j = 0, \ldots, n$ such that $G^{(j)} = \langle g_1^{(j)}, \ldots g_k^{(j)} \rangle$ where $k \in O(n)$. We can then use this to test whether G is solvable, as follows. If $g_1^{(n)}, \ldots, g_k^{(n)}$ are all equal to the identity element 1, then we have $G^{(n)} = \{1\}$ and so G is solvable. On the other hand, assume for contradiction that G is solvable but not all of $g_1^{(n)}, \ldots, g_k^{(n)}$ are equal to 1, *i.e.* $G^{(n)} \neq \{1\}$. Since the chain $G^{(0)} \rhd G^{(1)} \rhd \ldots$ eventually has $G^{(N)} = \{1\}$ for some $N \geq n$, it follows that, for $1 \leq i \leq n+1$, $G^{(i)}$ is a *proper* subgroup of $G^{(i-1)}$; otherwise, if $G^{(i)} = G^{(i-1)}$ for some such i, we would have

$$G^{(i-1)} = G^{(i)} = G^{(i+1)} = \ldots = G^{(n)} = \ldots = G^{(N)} = \{1\},$$

a contradiction since $G^{(n)} \neq \{1\}$. By Lagrange's theorem, $|G^{(i)}|$ divides $|G^{(i-1)}|$, so $|G^{(i)}| \leq 2 \cdot |G^{(i-1)}|$ since $|G^{(i)}| \neq |G^{(i-1)}|$. Then we have

$$|G| = |G^{(0)}| \geq 2 \cdot |G^{(1)}| \geq 4 \cdot |G^{(2)}| \geq \ldots \geq 2^{n+1}|G^{(n+1)}| \geq 2^{n+1}.$$

But this is impossible: since each element of G is uniquely encoded as a binary string of length n, there are at most 2^n distinct elements of G. We conclude that if G is solvable, it must be the case that all of $g_1^{(n)}, \ldots, g_k^{(n)}$ are equal to 1. Hence, given the elements $g_i^{(j)}$ as above, G is solvable if and only if all of $g_1^{(n)}, \ldots, g_k^{(n)}$ are equal to 1.

Now, under the assumption that G is solvable, we relabel the elements

$$g_1^{(n-1)}, \ldots, g_k^{(n-1)}, g_1^{(n-2)}, \ldots, g_k^{(n-2)}, \ldots, g_1^{(0)}, \ldots, g_k^{(0)}$$

as $g_1, \ldots, g_{kn}$ in the order given. For $1 \leq j \leq kn$, let $H_j = \langle g_1, \ldots, g_j \rangle$. Observe that

$$\begin{aligned} H_{kj} &= \langle g_1^{(n-1)}, \ldots, g_k^{(n-1)}, g_1^{(n-2)}, \ldots, g_k^{(n-2)} \ldots, g_1^{(n-j)}, \ldots, g_k^{(n-j)} \rangle \\ &= \langle G^{(n-1)} \cup G^{(n-2)} \cup \ldots \cup G^{(n-j)} \rangle \\ &= G^{(n-j)} \end{aligned}$$

for each $j = 1, \ldots, n$. As noted earlier, for an arbitrary group K we have $K' \lhd K$ and that K/K' is abelian. Then in particular, since $H_{kj} = G^{(n-j)}$ and $H_{k(j+1)} = G^{(n-j-1)}$, we have $H_{kj} \lhd H_{k(j+1)}$ with $H_{k(j+1)}/H_{kj}$ abelian, for all j. It then follows that

$$\{1\} =: H_0 \lhd H_1 \lhd \ldots \lhd H_{kn} = G.$$

Moreover, it is easy to see that for $1 \leq j \leq kn$ we have that

$$H_{j+1}/H_j = \langle g_1, \ldots, g_j, g_{j+1} \rangle / \langle g_1, \ldots, g_j \rangle$$

is cyclic, generated by the coset $g_{j+1}\langle g_1, \ldots, g_j \rangle = g_{j+1}H_j$. Recall that the objective of the algorithm is to compute the order $|G|$ of G. Thus far, we have efficiently found a chain $\{1\} = H_0 \lhd H_1 \lhd \ldots \lhd H_{kn} = G$ of subgroups of G. Suppose we knew the order of each factor group H_j/H_{j-1}. Then we could immediately compute $|G|$, since

$$\begin{aligned} |G| &= |H_{kn}| \\ &= \frac{|H_{kn}|}{|H_{kn-1}|} \frac{|H_{kn-1}|}{|H_{kn-2}|} \cdots \frac{|H_1|}{|H_0|} \\ &= |H_{kn}/H_{kn-1}| \cdot |H_{kn-1}/H_{kn-2}| \cdots |H_1/H_0|. \end{aligned}$$

Now, as previously noted, each factor group H_{j+1}/H_j is cyclic, generated by $g_{j+1}H_j$. The order of H_{j+1}/H_j is therefore the order of $g_{j+1}H_j$, hence the smallest positive integer r such that $g_{j+1}^r H_j = H_j$ or, equivalently, such that $g_{j+1}^r \in H_j$. In general, if

H is a subgroup of a group G and $g \in G$, we define the *order of g with respect to H* to be

$$r_H(g) := \min\{r \in \mathbb{Z} : r > 0, g^r \in H\}.$$

Note that this is well-defined, since by Lagrange's theorem we have $g^{|G|} = 1 \in H$. In this notation, the orders $|H_{j+1}/H_j|$ we wish to compute are equal to $r_{H_j}(g_{j+1})$ for $j = 0, \ldots, kn-1$. Watrous offers a modification of Shor's order-finding algorithm which will, given $g \in G$ and several copies of the state

$$|H\rangle = \frac{1}{\sqrt{|H|}} \sum_{h \in H} |h\rangle\,,$$

efficiently compute the relative order $r_H(g)$. This algorithm is then employed to calculate the orders $|H_{j+1}/H_j| = r_{H_j}(g_{j+1})$ and hence the order

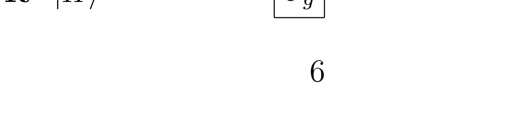
$$|G| = |H_{kn}/H_{kn-1}| \cdot |H_{kn-1}/H_{kn-2}| \cdots |H_1/H_0|.$$

3.1 Finding orders with respect to a given subgroup

Assume that we are given a finite black-box group G of encoding length n, a fixed element $g \in G$, an $n-$qubit quantum register $\mathbf{R}$ initialized to the state $|H\rangle$ for H a subgroup of G, and a quantum register $\mathbf{A}$ with basis $\{0, \ldots, N-1\}$ (where N is a parameter) initialized to the state $|0\rangle$. The algorithm provided for finding the order of g with respect to H is essentially a form of eigenvalue estimation. Let $r = r_H(g)$, and consider the cosets $H, gH, g^2H, \ldots, g^{r-1}H$. These cosets are distinct: suppose $i, j \in \{0, \ldots, r-1\}$ with $g^iH = g^jH$, and assume without loss of generality that $j \geq i$. Then $g^{j-i}H = H$, hence $g^{j-i} \in H$. But then either $j - i = 0$ or $r \leq j - i$ by definition of r, and since $i, j \in \{0, \ldots, r-1\}$ we have $r > j - i$, hence $i = j$. Since these cosets are disjoint, the corresponding states $\left|g^iH\right\rangle$ are orthogonal and therefore form an orthonormal basis for a subspace $\mathcal{W}$ of the $n-$qubit Hilbert space. Now, define the $n-$qubit gate U_g by linearly extending the action $|s\rangle \mapsto |gs\rangle$ for $s \in G$; this can be efficiently computed using the group oracle. This gate permutes the states $\left|g^iH\right\rangle$ since for $0 \leq i < r-1$ we have $U_g\left|g^iH\right\rangle = \left|g^{i+1}H\right\rangle$, and $U_g\left|g^{r-1}H\right\rangle = |g^rH\rangle = |H\rangle$ as $g^r \in H$. Therefore $\mathcal{W}$ is invariant under the action of U_g. Moreover, restricted to this subspace, U_g^r acts as the identity, since

$$U_g^r\left|g^iH\right\rangle = \left|g^rg^iH\right\rangle = \left|g^ig^rH\right\rangle = \left|g^iH\right\rangle.$$

It follows immediately that the eigenvalues of U_g, as an operator on $\mathcal{W}$, are of the form $e^{2\pi i\frac{k}{r}}$ for some $k \in \{0, \ldots, r-1\}$, since the minimal polynomial of U_g must divide $x^r - 1$. As with other examples of eigenvalue estimation, we define a controlled version of the gate U_g by linearly extending the action $|a\rangle|h\rangle \mapsto |a\rangle|g^ah\rangle$ for $a \in \{0, \ldots, N-1\}$ and $h \in G$. We perform eigenvalue estimation for U_g by executing the following circuit on the two registers:



A circuit diagram for computing $r_H(g)$

After the first $QFT_N \otimes I$ gate, the state of the system is

$$\frac{1}{\sqrt{N}} \sum_{a=0}^{N-1} |a\rangle\, |H\rangle \quad = \quad \frac{1}{\sqrt{N|H|}} \sum_{a=0}^{N-1} \sum_{h \in H} |a\rangle\, |h\rangle\, .$$

Next, the controlled U_g gate evolves the state to

$$\frac{1}{\sqrt{N|H|}} \sum_{a=0}^{N-1} \sum_{h \in H} |a\rangle\, |g^a h\rangle \quad = \quad \frac{1}{\sqrt{N}} \sum_{a=0}^{N-1} |a\rangle\, |g^a H\rangle\, .$$

We claim that the state resulting from the application of the final $QFT_N^{-1} \otimes I$ gate is

$$\frac{1}{N} \sum_{a=0}^{N-1} \sum_{b=0}^{N-1} e^{\frac{-2\pi i}{N} ab} |b\rangle\, |g^a H\rangle\, .$$

To see this, observe that

$$\begin{aligned}
(QFT_N \otimes I) \frac{1}{N} \sum_{a=0}^{N-1} \sum_{b=0}^{N-1} e^{\frac{-2\pi i}{N} ab} |b\rangle\, |g^a H\rangle \quad &= \quad \frac{1}{N} \sum_{a=0}^{N-1} \sum_{b=0}^{N-1} e^{\frac{-2\pi i}{N} ab} (\sum_{y=0}^{N-1} \frac{1}{\sqrt{N}} e^{\frac{2\pi i}{N} by} |y\rangle) |g^a H\rangle \\
&= \quad \frac{1}{N^{3/2}} \sum_{a=0}^{N-1} \sum_{b=0}^{N-1} \sum_{y=0}^{N-1} e^{\frac{-2\pi i}{N} b(y-a)} |y\rangle\, |g^a H\rangle\, .
\end{aligned}$$

For fixed a and y with $a \neq y$, the amplitude of $|y\rangle\, |g^a H\rangle$ in this state is

$$\begin{aligned}
\frac{1}{N^{3/2}} \left(\sum_{b=0}^{N-1} e^{\frac{2\pi i}{N}(y-a)b} \right) \quad &= \quad \frac{1}{N^{3/2}} \left(\frac{1 - e^{\frac{2\pi i}{N}(y-a)N}}{1 - e^{\frac{2\pi i}{N}(y-a)}} \right) \\
&= \quad \frac{1}{N^{3/2}} \left(\frac{1-1}{1 - e^{\frac{2\pi i}{N}(y-a)}} \right) \\
&= \quad 0,
\end{aligned}$$

whereas the amplitude of $|a\rangle\, |g^a H\rangle$ is $\frac{1}{N^{3/2}} \sum_{b=0}^{N-1} e^0 = \frac{1}{\sqrt{N}}$. Then we have

$$\frac{1}{\sqrt{N}} \sum_{a=0}^{N-1} |a\rangle\, |g^a H\rangle = (QFT_N \otimes I) \frac{1}{N} \sum_{a=0}^{N-1} \sum_{b=0}^{N-1} e^{\frac{-2\pi i}{N} ab} |b\rangle\, |g^a H\rangle$$

and hence

$$(QFT_N^{-1} \otimes I) \frac{1}{\sqrt{N}} \sum_{a=0}^{N-1} |a\rangle\, |g^a H\rangle = \frac{1}{N} \sum_{a=0}^{N-1} \sum_{b=0}^{N-1} e^{\frac{-2\pi i}{N} ab} |b\rangle\, |g^a H\rangle$$

as desired.

As in other applications of eigenvalue estimation, measuring the register $\mathbf{A}$ yields $b \in \{0, \dots, N-1\}$ such that, with high probability, $\frac{b}{N}$ is a good approximation to $\frac{k}{r}$ for some k, and taking N sufficiently large we can find r with high probability using the method of continued fractions.

Observe that, in the above method for calculating $r = r_H(g)$, we assumed that we had access to several copies of the uniform superposition $|H\rangle$ of the elements of H. We must therefore provide a method for efficiently constructing copies of the states $|H_j\rangle$ if we are to use the above for computing $|H_{j+1}/H_j| = r_{H_j}(g_{j+1})$ as we require.

3.2 Constructing uniform superpositions over subgroups

Suppose we are given a subgroup H of G and an element $g \in G$ such that $gH = Hg$. Then the subset $\langle g\rangle H = \{g^i h : i \in \mathbb{Z}, h \in H\}$ is a subgroup of G. To see this, we first note that $\langle g\rangle H$ contains the identity element $1 = g^0 1$. Next, let $g^{i_1}h_1, g^{i_2}h_2 \in \langle g\rangle H$. Since $gH = Hg$, it follows that $h_1 g^{i_2} = g^{i_2} h_3$ for some $h_3 \in H$, hence

$$\begin{aligned}(g^{i_1}h_1)(g^{i_2}h_2) &= g^{i_1}(h_1 g^{i_2})h_2 \\ &= g^{i_1}(g^{i_2}h_3)h_2 \\ &= g^{i_1+i_2}h_3h_2 \in \langle g\rangle H\end{aligned}$$

and so $\langle g\rangle H$ is closed under the group operation. Finally, we show that $\langle g\rangle H$ contains the inverse of each of its elements. For each $g^i h \in \langle g\rangle H$, the inverse of this element is $h^{-1}g^{-i}$ since

$$(g^i h)(h^{-1}g^{-i}) = g^i(hh^{-1})g^{-i} = g^i g^{-i} = 1.$$

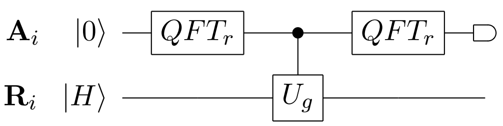
Since $Hg = gH$, it follows that there exists $h' \in H$ such that $h^{-1}g^{-i} = g^{-i}h' \in \langle g\rangle H$. Then indeed $\langle g\rangle H$ is a group. Moreover, it is easy to show that $H \triangleleft \langle g\rangle H$.

Now, for such H and g with $gH = Hg$, and given $r = r_H(g)$ along with a sufficiently large number of copies of the state $|H\rangle$, Watrous provides an algorithm for constructing several copies of the state $|\langle g\rangle H\rangle$. In particular, this can be used to inductively construct the states $|H_j\rangle$, as follows. Observe that for each j, since

$$\langle g_1, \dots, g_j\rangle = H_j \triangleleft H_{j+1} = \langle g_1, \dots, g_{j+1}\rangle$$

we have that $g_{j+1}H_j = H_j g_{j+1}$ and so $H_{j+1} = \langle g_{j+1}\rangle H_j$, so given $r_{H_j}(g_{j+1})$ and enough copies of $|H_j\rangle$ we can use the new method to construct several copies of $|H_{j+1}\rangle$, with which we can compute $r_{H_{j+1}}(g_{j+2})$ using the previous algorithm, and so on. We can easily construct as many copies of the state $|H_0\rangle = |1\rangle$ as we would like, so if we construct enough of these and compute the order $r_{H_0}(g_1)$ at the outset, we can thereby compute all of the orders $|H_{j+1}/H_j| = r_{H_j}(g_{j+1})$ which we require. The new algorithm proceeds as follows:

Assume that we have registers $\mathbf{R}_1, \dots, \mathbf{R}_l$ initialized to the state $|H\rangle$, and registers $\mathbf{A}_1, \dots, \mathbf{A}_l$ each with the basis $\{0, \dots, r-1\}$. For each $i \in \{1, \dots, l\}$, execute the circuit



which, by a straightforward computation, brings the state of the pair $(\mathbf{A}_i, \mathbf{R}_i)$ to

$$\frac{1}{r}\sum_{a_i=0}^{r-1}\sum_{b_i=0}^{r-1} e^{\frac{2\pi i}{r}a_i b_i} |b_i\rangle |g^{a_i}H\rangle$$

prior to measurement. Let b_i be the result of the measurement of $\mathbf{A}_i$. Denoting the new state of $\mathbf{R}_i$ by $|\psi_i\rangle$, we have

$$|\psi_i\rangle = \frac{1}{\sqrt{r}}\sum_{a_i=0}^{r-1} e^{\frac{2\pi i}{r}a_i b_i} |g^{a_i}H\rangle\,.$$

For each i, the probability that b_i is coprime to r is equal to $\frac{\varphi(r)}{r}$, where φ is the *Euler totient* function which returns the number of positive integers that are less than and coprime to its argument. By a result from classical number theory [7], there exists a positive constant δ such that $\frac{\varphi(r)}{r} > \frac{\delta}{\log\log r}$, and thus we can take $l \in O(\log\log r)$ so that, with high probability, there is some $k \in \{0, \ldots, r-1\}$ such that b_k is coprime to r. We proceed to use $|\psi_k\rangle$ to convert the other states $|\psi_i\rangle$ into $|\langle g\rangle H\rangle$. Let $i \in \{0, \ldots, r-1\}$ with $i \neq k$. Since b_k is coprime to r, there exist integers x, y such that $xb_k + yr = 1$. Let $c = xb_i$, and reversibly multiply the contents of $\mathbf{R}_k$ by f^c where f is the group element contained in $\mathbf{R}_i$. Before this multiplication, the state of the pair $(\mathbf{R}_i, \mathbf{R}_k)$ is

$$\begin{aligned}
|\psi_i\rangle |\psi_k\rangle &= \frac{1}{\sqrt{r}}\sum_{a_i=0}^{r-1} e^{\frac{2\pi i}{r}a_i b_i} |g^{a_i}H\rangle |\psi_k\rangle \\
&= \frac{1}{\sqrt{r|H|}}\sum_{a_i=0}^{r-1}\sum_{h\in H} e^{\frac{2\pi i}{r}a_i b_i} |g^{a_i}h\rangle |\psi_k\rangle
\end{aligned}$$

and hence, after the multiplication, the state of the pair is

$$\frac{1}{\sqrt{r|H|}}\sum_{a_i=0}^{r-1}\sum_{h\in H} e^{\frac{2\pi i}{r}a_i b_i} |g^{a_i}h\rangle\, U_{(g^{a_i}h)^c} |\psi_k\rangle = \frac{1}{\sqrt{r|H|}}\sum_{a_i=0}^{r-1}\sum_{h\in H} e^{\frac{2\pi i}{r}a_i b_i} |g^{a_i}h\rangle\, (U_{g^{a_i}h})^c |\psi_k\rangle$$

(Recall that, for $t \in G$, the gate U_t acts by left-multiplying by t.) Since $gH = Hg$, for each $h \in H$ there exists $h' \in H$ such that $g^{a_i}h = h'g^{a_i}$. Then

$$\begin{aligned}
U_{g^{a_i}h}\left|\psi_k\right\rangle &= \frac{1}{\sqrt{r}}\sum_{a_k=0}^{r-1} e^{\frac{2\pi i}{r}a_k b_k}\left|g^{a_i}hg^{a_k}H\right\rangle \\
&= \frac{1}{\sqrt{r}}\sum_{a_k=0}^{r-1} e^{\frac{2\pi i}{r}a_k b_k}\left|g^{a_i}g^{a_k}h'H\right\rangle \\
&= \frac{1}{\sqrt{r}}\sum_{a_k=0}^{r-1} e^{\frac{2\pi i}{r}a_k b_k}\left|g^{a_i+a_k}H\right\rangle \\
&= \frac{1}{\sqrt{r}}\sum_{a_k=0}^{r-1} e^{\frac{2\pi i}{r}(a_k-a_i) b_k}\left|g^{a_k}H\right\rangle \\
&= e^{\frac{-2\pi i}{r}a_i b_k}\left|\psi_k\right\rangle .
\end{aligned}$$

Then the state of the pair is

$$\begin{aligned}
\frac{1}{\sqrt{r|H|}}\sum_{a_i=0}^{r-1}\sum_{h\in H} e^{\frac{2\pi i}{r}a_i b_i}\left|g^{a_i}h\right\rangle e^{\frac{-2\pi i}{r}a_i b_k c}\left|\psi_k\right\rangle &= \frac{1}{\sqrt{r|H|}}\sum_{a_i=0}^{r-1}\sum_{h\in H} e^{\frac{2\pi i}{r}a_i b_i}\left|g^{a_i}h\right\rangle e^{\frac{-2\pi i}{r}a_i b_i x b_k}\left|\psi_k\right\rangle \\
&= \frac{1}{\sqrt{r|H|}}\sum_{a_i=0}^{r-1}\sum_{h\in H} e^{\frac{2\pi i}{r}a_i b_i}\left|g^{a_i}h\right\rangle e^{\frac{-2\pi i}{r}a_i b_i(1-yr)}\left|\psi_k\right\rangle \\
&= \frac{1}{\sqrt{r|H|}}\sum_{a_i=0}^{r-1}\sum_{h\in H} e^{\frac{2\pi i}{r}a_i b_i}\left|g^{a_i}h\right\rangle e^{\frac{-2\pi i}{r}a_i b_i)}\left|\psi_k\right\rangle \\
&= \frac{1}{\sqrt{r|H|}}\sum_{a_i=0}^{r-1}\sum_{h\in H}\left|g^{a_i}h\right\rangle\left|\psi_k\right\rangle \\
&= \left|\langle g\rangle H\right\rangle\left|\psi_k\right\rangle
\end{aligned}$$

Repeating this for each $i \neq k$, the result is $l-1$ copies of the state $|\langle g\rangle H\rangle$ as desired. By preparing a sufficient number of copies of the initial state $|H_0\rangle$, using these algorithms allows for the order of G to be efficiently computed as described.

4 Applications

As previously mentioned, several group-theoretic problems can be reduced to computing orders of subgroups, given a list of generators. We describe a few such problems here.

1. (Membership testing.) Suppose we are given $g_1, \ldots, g_k, h \in G$ where $\langle g_1, \ldots, g_k, h\rangle$ is solvable. (This latter condition holds, for example, if G itself is solvable.) We can use Watrous's algorithm to determine whether $h \in \langle g_1, \ldots, g_k\rangle$ in quantum polynomial time: $h \in \langle g_1, \ldots, g_k\rangle$ if and only if $\langle g_1, \ldots, g_k\rangle = \langle g_1, \ldots, g_k, h\rangle$, which is equivalent to $|\langle g_1, \ldots, g_k\rangle| = |\langle g_1, \ldots, g_k, h\rangle|$. Therefore, computing and comparing the orders of these two subgroups will permit us to verify the membership.

2. (Subgroup testing.) Suppose we are given $g_1, \ldots, g_k, h_1, \ldots, h_l \in G$ such that both $\langle g_1, \ldots, g_k \rangle$ and $\langle h_1, \ldots, h_l \rangle$ are solvable. We can efficiently determine whether $\langle g_1, \ldots, g_k \rangle$ is a subgroup of $\langle h_1, \ldots, h_l \rangle$ by testing whether all g_i are members of $\langle h_1, \ldots, h_l \rangle$, using the method of membership testing above. Observe that this also allows us to determine whether $\langle g_1, \ldots, g_k \rangle = \langle h_1, \ldots, h_l \rangle$ by testing whether each is a subgroup of the other.

3. (Normality testing.) Given $g_1, \ldots, g_k, h_1, \ldots, h_l \in G$ such that both $\langle g_1, \ldots, g_k \rangle$ and $\langle h_1, \ldots, h_l \rangle$ are solvable, we can test whether $\langle g_1, \ldots, g_k \rangle \lhd \langle h_1, \ldots, h_l \rangle$ by checking whether $\langle g_1, \ldots, g_k \rangle$ is a subgroup of $\langle h_1, \ldots, h_l \rangle$, then testing whether $h_j^{-1} g_i h_j \in \langle g_1, \ldots, g_k \rangle$ for all $i \in \{1, \ldots, k\}$ and $j \in \{1, \ldots, l\}$.

4. (Homomorphisms.) Let G and H be groups. The *direct product* of G and H is the cartesian product $G \times H$, equipped with the group operation given by $(g_1, h_1)(g_2, h_2) = (g_1 g_2, h_1 h_2)$, and a function $\phi : G \to H$ is a *homomorphism* if $\phi(st) = \phi(s)\phi(t)$ for all $s, t \in G$. Suppose that $G = \langle g_1, \ldots, g_k \rangle$ and H are solvable, and we are given a function $\hat{\phi} : \{g_1, \ldots, g_k\} \to H$ which can be efficiently computed. It can be shown [6] that $\hat{\phi}$ extends to a homomorphism $\phi : G \to H$ if and only if the subgroup $\langle (g_1, \hat{\phi}(g_1)), \ldots, (g_k, \hat{\phi}(g_k)) \rangle$ of $G \times H$ is of the same order as G, and this can be tested in quantum polynomial time since both of these groups are solvable.

5. (Verification of kernels.) Let $\phi : G \to H$ be a homomorphism. The *kernel* of ϕ is defined by $\ker \phi = \{g \in G : \phi(g) = 1\}$, and is a normal subgroup of G. Assume we are given solvable G and H, and a method for efficiently computing such ϕ. Given a subgroup N of G defined by a list of generators $n_1, \ldots, n_k$, it can be shown [6] that $N = \ker \phi$ if and only if $\phi(n_i) = 1$ for each i and $|N| = |G| / |\langle \phi(n_1), \ldots, \phi(n_k) \rangle|$. These orders can be computed in quantum polynomial time using Watrous's algorithm.

5 Conclusion

Watrous has given a quantum algorithm for computing the order of a finite solvable black-box group, producing a uniform superposition over its elements as a byproduct, in polynomial time. Furthermore, this algorithm can be used to solve several other group-theoretic problems that reduce to computing orders of certain groups. Watrous mentions two other problems for solvable black-box groups for which, at the time of publication, no polynomial-time algorithms were known. The Group Intersection problem is to determine, given elements $g_1, \ldots, g_k$ and $h_1, \ldots, h_l$ of a solvable black-box group G, whether $\langle g_1, \ldots, g_k \rangle$ and $\langle h_1, \ldots, h_l \rangle$ have any elements in common other than 1. Similarly, the Coset Intersection problem is to determine whether a given coset of $\langle g_1, \ldots, g_k \rangle$ has nonempty intersection with $\langle h_1, \ldots, h_l \rangle$. A recent paper by Fenner and Zhang [8] shows the existence of quantum polynomial-time algorithms for these two problems, provided that the underlying solvable groups satisfy an additional "smoothness" criterion. Other directions for research include discovering methods for similar problems in non-solvable groups, possibly by extending known methods for solvable groups to certain larger classes of groups.

References

[1] J. Watrous. Quantum algorithms for solvable groups. In *Proceedings of the 33rd Annual ACM Symposium on Theory of Computing*, 60-67, 2001.

[2] L. Babai and E. Szemerédi. On the complexity of matrix group problems I. In *Proceedings of the 25th Annual Symposium on Foundations of Computer Science*, 229-240, 1984.

[3] V. Arvind and N. V. Vinodchandran. Solvable black-box group problems are low for PP. *Theoretical Computer Science*, 180:17-45, 1997.

[4] D. S. Dummit and R. M. Foote. *Abstract Algebra*. Wiley, third edition, 2004.

[5] L. Babai, G. Cooperman, L. Finkelstein, E. Luks, and Á. Seress. Fast Monte Carlo algorithms for permutation groups. *Journal of Computer and System Sciences*, 50:296-307, 1995.

[6] L. Babai. Bounded round interactive proofs in finite groups. *SIAM Journal on Discrete Math*, 5(1):88-111, 1992.

[7] G. H. Hardy and E. M. Wright. *An Introduction to the Theory of Numbers*. Oxford University Press, fifth edition, 1979.

[8] S. Fenner and Y. Zhang. Quantum algorithms for a set of group theoretic problems. *Theoretical Computer Science*, 215-227, 2007.