A hidden shift quantum algorithm

2000 J. Phys. A: Math. Gen. 33 8973

(http://iopscience.iop.org/0305-4470/33/48/325)

J. Phys. A: Math. Gen. **33** (2000) 8973–8979. Printed in the UK PII: S0305-4470(00)14016-8

A hidden shift quantum algorithm

J Twamley

Department of Mathematical Physics, National University of Ireland, Maynooth, Maynooth, County Kildare, Republic of Ireland

E-mail: jtwamley@thphys.may.ie

Received 17 May 2000, in final form 26 October 2000

Abstract. We examine the application of quantum algorithms to the non-Abelian hidden subgroup problem and focus on the dihedral hidden subgroup problem (DHSP). Ettinger and Høyer have recently discovered an algorithm which, although efficient in the number of operations of the quantum computation, requires classical post-processing which grows exponentially with the size of the input. We first show that the DHSP can be reduced to another problem: how to efficiently estimate an unknown shift k, in a one-to-one map $f : \mathbb{Z}_M \to \mathbb{Z}_M$, given two oracles which provide $x \to f(x)$, and $x \to f(x \oplus k)$, with k and f unknown. We devise an algorithm which uses amplitude amplification to obtain an estimate for the unknown shift k in a number $\mathcal{O}(\sqrt{M})$ oracle calls.

1. Introduction

All of the known quantum algorithms which run exponentially faster than their most efficient probabilistic classical counterparts can be reformulated as algorithms which solve particular cases of the *hidden subgroup problem* (HSP for short), or *unknown subgroup problem* [1–3]. Essentially, this problem amounts to finding the generators of an unknown subgroup $\mathcal{K}$ of a group $\mathcal{G}$, given a function $f : \mathcal{G} \to \mathcal{R}$, which is constant and distinct on the cosets of $\mathcal{K}$ in $\mathcal{G}$, and where $\mathcal{R}$ is a finite set. Almost all quantum algorithms which are exponentially faster than the most efficient classical probabilistic algorithm are examples of an Abelian hidden subgroup problem where the group $\mathcal{G}$ (and thus $\mathcal{K}$) are Abelian. In particular, Shor's factorization problem corresponds to finding the generators of the unknown additive subgroup $\mathcal{K} = r\mathbb{Z}$, of integer multiples of r, where r is the order of of an element a from the group of integers modulo N, and $\mathcal{G} = \mathbb{Z}$. Here the function f maps $x \to a^x \bmod N$. In fact, one can map Deutsch's problem, Simon's problem, the discrete logarithm problem and the Abelian stabilizer problem onto Abelian hidden subgroup problems (see [4] and [6] for details). It is now known how to efficiently solve any Abelian HSP using a quantum algorithm. The next generalization, using quantum algorithms to efficiently solve a non-Abelian HSP, remains, except for one particular case [5], an open problem. The motivations to expand the known techniques to non-Abelian situations are high as it is known that the difficult problem of graph isomorphism is equivalent to the non-Abelian HSP for symmetric groups.

In this paper we concentrate on the dihedral hidden subgroup problem (DHSP), where $\mathcal{G} = D_N$, and where D_N is the Nth dihedral group or the symmetry group of the N-sided polygon. The hidden subgroup problem for D_N has been considered by Ettinger and Høyer in [1]. They found a quantum algorithm which, given a function $f : \mathcal{G} \to \mathcal{R}$, and which satisfies the dihedral hidden subgroup promise with respect to some subgroup $\mathcal{K}$, there exists

0305-4470/00/488973+07$30.00

a quantum algorithm which uses $\mathcal{O}(\log N)$ evaluations of f, and outputs a subset $X \subseteq D_N$ such that X is a generating set for $\mathcal{K}$ with probability at least $1 - 2/N$. The dihedral subgroup promise is the statement that f is distinct and constant on the left cosets of $\mathcal{K}$ in $\mathcal{G}$. Although this appears, at first, to solve the DHSP, the authors point out that to extract a good estimate for the generating set X, from the output of the quantum algorithm, requires an exponential amount of classical post-processing and thus the combined quantum and classical processing is exponential in $\log N$.

In the following we show that the DHSP can be recast into another, more tractable, problem which we will call the hidden shift problem. For the simplest case, we show that this problem is, in fact, Deutsch's problem, and thus possesses a quantum algorithmic solution. We then explore various approaches one could take towards finding a more general solution to the quantum hidden shift problem. We find that the technique of amplitude amplification [6], points the way to a solution and we give the details of the quantum algorithm which solves the hidden shift problem and thus the DHSP.

2. Dihedral HSP and the hidden shift problem

We begin by reducing the DHSP to the hidden shift problem. We first note, along with [1], that the group D_N is isomorphic to the group $\mathbb{Z}_N \rtimes_\phi \mathbb{Z}_2$, where the multiplication is defined by $(a_1, b_1)(a_2, b_2) = (a_1 + \phi(b_1)(a_2), b_1 + b_2)$, and where the homomorphism $\phi : \mathbb{Z}_2 \to \mathrm{Aut}(\mathbb{Z}_N)$ is defined by $1 \to \phi(1)(a) = -a$. We take a group element in D_N to be represented by the double (a, b), in $\mathbb{Z}_N \rtimes_\phi \mathbb{Z}_2$. It was shown in [1], through an analysis of the group structure of D_N, that the set X of generators for the hidden subgroup $\mathcal{K}$ can be made up of two parts. The first part, X_1, which generates the portion of the hidden subgroup when the map f is restricted to the Abelian subgroup of D_N, can be found using standard quantum algorithmic techniques for solving the Abelian HSP. In [1] they then consider the restriction f_M of f to $D_M = D_N/\langle X_1 \rangle \to \mathcal{R}$. This restricted map is also subject to a hidden subgroup problem with the hidden subgroup being either $\mathcal{K}_2 = \{(0, 0)\}$ or $\mathcal{K}_2 = \{(0, 0), (k_0, 1)\}$, with the final complete generating set being $X = X_1$ or $X = X_1 \cup \{(k_0, 1)\}$, respectively. The trivial case where $\mathcal{K}_2 = \{(0, 0)\}$, can be dealt with separately and solved. For the non-trivial case, the task of determining the generating set for the hidden subgroup is thus reduced to finding the value of k_0. In the appendix we show that this task is identical to the following problem: one is given two one-to-one and *unknown* functions (or oracles), f_0 and f_1, which map $\mathbb{Z}_M \to \mathbb{Z}_M$, but which satisfy the *promise*, $f_1(a) = f_0(a + k_0)$, $a \in \mathbb{Z}_M$, where k_0 is again *unknown* (see the appendix). In [1] the quantum oracle representing the map $f : D_N \to \mathcal{R}$, for the DHSP is given by

$$\hat{U}_f : |A, b, c\rangle \to |A, b, f(A, b) \oplus c\rangle \tag{1}$$

where $a \in \mathbb{Z}_N$ and $b \in \mathbb{Z}_2$, while the oracle f_M which acts on D_M can be viewed in terms of the states

$$|a, b, c\rangle = \frac{1}{\sqrt{|\langle X_1 \rangle|}} \sum_{x \in \langle X_1 \rangle} |a \oplus x, b, c\rangle \tag{2}$$

where now we can consider $a \in \mathbb{Z}_M$, $M = \min\{1 \leqslant j \leqslant N \mid (j, 0) \in \langle X_1 \rangle\}$, and where the addition in the right-hand side of (2) is taken modulo N. The 'reduced' quantum oracle representing the map $f_M : D_M \to \mathcal{R}$ can be written as

$$\hat{U}_{\mathrm{DHSP}} : |a, b, c\rangle \to |a, b, f_M(a, b) \oplus c\rangle \equiv |a, b, f_b(a) \oplus c\rangle. \tag{3}$$

The task now is to estimate k_0.

3. Strategies towards a quantum algorithm

We begin by noting that the hidden shift problem, in the case $M = 2$, is in fact Deutsch's problem and thus possesses a quantum algorithmic solution. For $M > 2$, however, since the details of the function f_0 and f_1 are unknown, one might wish to encode into the algorithm the means of identifying and coping with all of the possible one-to-one maps $f_0 : \mathbb{Z}_M \to \mathbb{Z}_M$. This can be done for $M = 2$. However, as the number of such maps goes as M factorial, while the dimension of the associated Hilbert space goes as M, this cannot possibly succeed for $M > 2$. A strategy similar to the one taken in [1] would be to make use of the fact that a function whose argument suffers a constant shift when transformed by the Fourier transform picks up a phase which is proportional to this shift, relative to the Fourier transform of the unshifted function. One can design a quantum algorithm along this reasoning using the quantum Fourier transform. However, as in [1], although the quantum algorithm requires polynomial steps, one must estimate k_0 from a non-uniform probability distribution, and this again requires an exponential amount of classical post-processing.

From the results of the previous section, we have available to us two oracles, $a \to f_0(a)$, the unshifted function, and $a \to f_1(a) = f_0(a + k_0)$, the shifted function, with which to construct our quantum algorithm. It is not hard to show that if one were given, in addition to these two oracles, the oracles $a \to f_i(a) = f_0(a + 2^i k_0)$, $i = 1, 2, \ldots, (\log M) - 1$, then one could use quantum Fourier transform techniques to efficiently estimate k_0. However, the underlying problem, the DHSP, only yields to us the two oracles f_0 and f_1 (see the appendix). Thus, to solve the DHSP we must restrict ourselves to work with f_0 and f_1.

There are two possible classes of quantum algorithms which might be applied to solve this problem, namely algorithms based on Shor's factorization method or Grover's search method. The former will yield an exponential increase in speed, while the latter will yield a square-root increase in speed. We have so far been unsuccessful in discovering an exponentially efficient algorithm for solving the hidden shift problem. However, below we give an algorithm which uses amplitude amplification to obtain an estimate of the unknown shift k_0 in $\mathcal{O}(\sqrt{M})$ oracle calls. Such a procedure will entail more quantum oracle calls than the procedure advanced in [1]. Although calls to a quantum oracle may be costly to implement physically, the entire algorithm (which includes the quantum and classical post-processing) presented here will scale more efficiently than in [1].

3.1. Grover's search or amplitude amplification

The Grover operator $\hat{Q}$ is given by [7]

$$\hat{Q} = -\hat{H}\hat{I}_0\hat{H}\hat{I}_{x_0} \tag{4}$$

where $\hat{H}$ is the M-dimensional Hadamard operator and $\hat{I}_0$ and $\hat{I}_{x_0}$ are inversion operators that we will define below. The Grover operator acts within an M-dimensional space. However, the algorithm described below will require the use of several 'scratch' registers to effect the operator $\hat{I}_{x_0}$. These scratch registers are initialized to zero and are returned to zero unitarily during the implementation of $\hat{I}_{x_0}$. The starting state of the algorithm is the uniform state $|\psi_i\rangle = \hat{H}|0\rangle$, where $|0\rangle$ is the zero state of an M-qubit register, a register which we shall denote as the *index register*. The operator $\hat{I}_0$ reflects any state in this register through the zero state, i.e.

$$\hat{I}_0 = \hat{\mathbb{I}}_h - 2|0\rangle_{II}\langle 0| \tag{5}$$

where we have inserted a subscript I to indicate that we are working in the Hilbert space of the index register. The operator $\hat{I}_{x_0}$ will be built to effect the following transformation:

$$\hat{I}_{x_0}|h\rangle_I = \begin{cases} +|h\rangle_I & h \neq k_0 \\ -|h\rangle_I & h = k_0 \end{cases} \tag{6}$$

or

$$\hat{I}_{x_0} = \hat{\mathbb{I}}_I - 2|k_0\rangle_{II}\langle k_0| \tag{7}$$

where k_0 is the unknown shift in the shifted oracle $f_1(a) = f_0(a + k_0)$.

Once equipped with $\hat{Q}$ we iterate $\hat{Q}^n|\psi_i\rangle_I$, approximately $n = \sqrt{M}$ times. From [6], since there is only one 'marked card' in the index list, when $h = k_0$, we obtain

$$|k_0\rangle_I \approx \hat{Q}^{\sqrt{M}}|\psi_i\rangle_I. \tag{8}$$

One can thus determine the value of k_0 with near unit probability by measuring the final state of the index register.

3.2. Constructing $\hat{I}_{x_0}$

To effect the transformation (6), we use extra 'scratch' quantum registers. We introduce five extra scratch registers, which are all (except one) M-qubit registers initialized to the zero state. We label these registers as the x-register $|0\rangle_x$, the oracle register $|0\rangle_\gamma$, the unshifted register, $|0\rangle_0$, the shifted register $|0\rangle_1$, and the difference register, $|0\rangle_\Delta$. The oracle register is a single-qubit register. The following steps will make use of the oracle (3), a single-qubit NOT gate, $\hat{U}_{\rm NOT}$, and standard addition and subtraction gates, $\hat{U}_\pm : |a, b\rangle \to |a, b \pm a\rangle$. The registers acted upon by these gates will be denoted as superscripts, i.e. the addition of the index register to the difference register will be given by $\hat{U}_+^{h\Delta}$.

Step 1. The initial state of the index and scratch registers is thus taken to be

$$|\Psi_i\rangle = \frac{1}{\sqrt{M}} \sum_{h=0}^{M-1} |h\rangle_I \otimes |0\rangle_x \otimes |0\rangle_\gamma \otimes |0\rangle_0 \otimes |0\rangle_1 \otimes |0\rangle_\Delta \tag{9}$$

or

$$|\Psi_i\rangle = \frac{1}{\sqrt{M}} \sum_{h=0}^{M-1} |h, 0, 0, 0, 0, 0\rangle \tag{10}$$

where it is understood that the order of the registers is *index*, x, *oracle, unshifted, shifted* and *difference*.

Step 2. We next apply an M-dimensional Hadamard operator in the x-register space, $\hat{H}^x$, to obtain

$$|\Psi_2\rangle = \frac{1}{M} \sum_{h=0}^{M-1} \sum_{x=0}^{M-1} |h, x, 0, 0, 0, 0\rangle. \tag{11}$$

Step 3. We next apply a bit flip, $\hat{U}_{\rm NOT}^\gamma$, targeting the oracle register and then apply the oracle (3), targeting a from the x register, b from the oracle register and c from the shifted register, $\hat{U}_{\rm DHSP}^{x\gamma 1}$. After this we again apply the bit flip to the oracle register to obtain

$$|\Psi_3\rangle = \frac{1}{M} \sum_{h=0}^{M-1} \sum_{x=0}^{M-1} |h, x, 0, 0, f_0(x \oplus k_0), 0\rangle. \tag{12}$$

Step 4. We next add the index register to the x-register by applying $\hat{U}_+^{hx}$, to obtain

$$|\Psi_4\rangle = \frac{1}{M}\sum_{h=0}^{M-1}\sum_{x=0}^{M-1}|h, x\oplus h, 0, f_0(x\oplus k_0), 0\rangle. \tag{13}$$

Step 5. We then apply the oracle (3) again, this time targeting c from the unshifted register by applying $\hat{U}_{\mathrm{DHSP}}^{x\gamma 0}$, to obtain

$$|\Psi_5\rangle = \frac{1}{M}\sum_{h=0}^{M-1}\sum_{x=0}^{M-1}|h, x\oplus h, 0, f_0(x\oplus h), f_0(x\oplus k_0), 0\rangle. \tag{14}$$

Step 6. We now undo the addition by applying $\hat{U}_+^{hx\,\dagger}$, to obtain

$$|\Psi_6\rangle = \frac{1}{M}\sum_{h=0}^{M-1}\sum_{x=0}^{M-1}|h, x, 0, f_0(x\oplus h), f_0(x\oplus k_0), 0\rangle. \tag{15}$$

Step 7. We now add the unshifted register to the difference register by applying $\hat{U}_+^{0\Delta}$, and then subtract the shifted register from the difference register by applying $\hat{U}_-^{1\Delta}$, to finally obtain

$$|\Psi_7\rangle = \frac{1}{M}\sum_{h=0}^{M-1}\sum_{x=0}^{M-1}|h, x, 0, f_0(x\oplus h), f_0(x\oplus k_0), f_0(x\oplus h) - f_0(x\oplus k_0)\rangle. \tag{16}$$

Since the function $f_0 : \mathbb{Z}_M \to \mathbb{Z}_M$ is one to one, the difference register will possess the zero value iff $h = k_0$. In fact, in the $M \times M$, superposition (16), all the M terms in the x sum which have $h = k_0$ in their index register will have a zero entry in their difference register. The evolution described from steps 2 to 9 can be given as

$$\hat{U}_P \equiv \hat{U}_-^{1\Delta}\,\hat{U}_+^{0\Delta}\,\hat{U}_+^{hx\,\dagger}\,\hat{U}_{\mathrm{DHSP}}^{x\gamma 0}\,\hat{U}_+^{hx}\,\hat{U}_{\mathrm{NOT}}^{\gamma}\,\hat{U}_{\mathrm{DHSP}}^{x\gamma 1}\,\hat{U}_{\mathrm{NOT}}^{\gamma}\,\hat{H}^x. \tag{17}$$

Step 8. To complete the process we now apply the operator $\hat{I}_0^{\Delta}$, where

$$\hat{I}_0^{\Delta} \equiv \mathbb{I}_I \otimes \mathbb{I}_x \otimes \mathbb{I}_\gamma \otimes \mathbb{I}_0 \otimes \mathbb{I}_1 \otimes (\mathbb{I}_\Delta - 2|0\rangle_{\Delta\Delta}\langle 0|). \tag{18}$$

This effects

$$\hat{I}_0^{\Delta}|h, x, 0, f_0(x\oplus h), f_0(x\oplus k_0), f_0(x\oplus h) - f_0(x\oplus k_0)\rangle \tag{19}$$

$$= \begin{cases} +|h, x, 0, f_0(x\oplus h), f_0(x\oplus k_0), f_0(x\oplus h) - f_0(x\oplus k_0)\rangle & h \neq k_0 \\ -|h, x, 0, f_0(x\oplus h), f_0(x\oplus k_0), 0\rangle & h = k_0. \end{cases} \tag{20}$$

Step 9. We now return the scratch registers to their initial state by applying $\hat{U}_P^{\dagger}$. After this, we finally return to the state

$$|\Psi_f\rangle = \hat{U}_P^{\dagger}\,\hat{I}_0^{\Delta}\,\hat{U}_P\,|\Psi_i\rangle \tag{21}$$

$$= \frac{1}{\sqrt{M}}\Bigg(\sum_{h=0, h\neq k_0}^{M-1}|h, 0, 0, 0, 0, 0\rangle - |k_0, 0, 0, 0, 0, 0\rangle\Bigg) \tag{22}$$

$$= \frac{1}{\sqrt{M}}\Bigg(\sum_{h=0, h\neq k_0}^{M-1}|h\rangle_I - |k_0\rangle_I\Bigg). \tag{23}$$

Thus, the operator, $\hat{I}_{x_0}$, can be broken down into $\hat{U}_P^\dagger\, \hat{I}_0^\Delta\, \hat{U}_P$. Since there is only one 'marked card', to amplitude-amplify the initial state $|\Psi_i\rangle_I$ to the final state $|k_0\rangle_I$ requires $\mathcal{O}(\sqrt{M})$ calls of the Grover iterate $\hat{Q}$. This would result in $\mathcal{O}(4\sqrt{M})$ oracle calls (one Grover step requires an oracle call for each of the steps 3 and 5 and two further calls to the oracles to undo these in step 9). However, this can be reduced to $\mathcal{O}(2\sqrt{M})$ calls by working in a larger Hilbert space and setting the initial state $|\Psi_i\rangle = |\Psi_3\rangle$. We then set $\hat{I}_{x_0} = \hat{U}_P'^\dagger\, \hat{I}_0^\Delta\, \hat{U}_P'$, where $\hat{U}_P' = \hat{U}_-^{1\Delta}\, \hat{U}_+^{0\Delta}\, \hat{U}_+^{hx\,\dagger}\, \hat{U}_{\mathrm{DHSP}}^{x\gamma 0}\, \hat{U}_+^{hx}$.

4. Conclusion

In this paper we have shown that the non-Abelian DHSP can be reduced to the hidden shift problem. Through the method of amplitude amplification we were able to devise an algorithm which produced a good estimate for the hidden shift and thus a solution for the DHSP. Although [1] found a quantum algorithm which only called the oracles $\sim\mathcal{O}(\log M)$ times (smaller than required here), the resulting classical information required an exponential amount of classical post-processing to arrive at an estimate for k_0. The algorithm outlined in this paper requires $\mathcal{O}(\sqrt{M})$ oracle calls. The discovery of a quantum algorithm which could estimate the hidden shift in $\mathcal{O}(\log M)$ oracle calls with a polynomial classical overhead would be quite significant. Perhaps, the insight afforded by the '*interference interpretation*' [2] of quantum algorithms might help towards such a discovery.

Acknowledgments

The author would like to thank M Mosca and T Hovland and the paper's referees for valuable discussions. Financial support for this work came from the British Council and the EU IST FET QIPC Project QIPDDF.

Appendix

We now outline how the problem of determining the value of k_0 in the DHSP is related to the hidden shift problem. In [1], the hidden subgroup $\mathcal{K}_2$ of D_M is either trivial, $\{(0,0)\}$, or is $\{(0,0),(k_0,1)\}$. The trivial case can be dealt with separately. For the non-trivial case, the hidden subgroup which returns all of the problem states that the restriction of the map f on the left cosets of $\mathcal{K}_2$ in D_M is constant and distinct. The order of D_M is $2M$; however, there are only M left cosets. These M cosets, C_i, $i \in [1,\ldots,M]$, can be labelled by the first index a,

$$C_a = \{(a,0),(a+k_0,1)\}. \tag{A1}$$

The function f, which is one to one, is constant and distinct on these cosets and thus $f((a,0)) = f((a+k_0,1)) \equiv f_a$. We can thus take the restriction of f acting on elements (a,b), where $b=0$, to be a new map (or oracle) $f_0 : \mathbb{Z}_M \to \mathbb{Z}_M : a \to f_0(a) \equiv f((a,0))$, while the restriction of f to the elements $(a,1)$ yields another oracle, $f_1 : \mathbb{Z}_M \to \mathbb{Z}_M : a \to f_1(a) = f_0(a+k_0) \equiv f((a,1))$. In the corresponding HSP, the function f and thus f_0 are not known in detail except that they are both one to one. The task is to determine the shift k_0. The situation can be understood in the following manner: you are given $a \in \mathbb{Z}_M$ and the image of an *unknown* one-to-one function $f_0 : \mathbb{Z}_M \to \mathbb{Z}_M$. You are then given the image of the same function, except now the domain has been shifted by an unknown amount k_0. Your task is to determine k_0. Classically, one could take a particular value of a, say a_1, determine $f_0(a_1)$, and

then begin searching through a_j, $j = 1, \ldots, M$, to find a value a_2, such that $f_1(a_2) = f_0(a_1)$. Then $a_2 = a_1 + k_0$. However, this search will take on average $\frac{1}{2}M$ iterations.

References

[1] Ettinger M and Høyer P 1999 *Proc. 16th Ann. Symp. on Theoretical Aspects of Computer Science (Trier) (Lecture Notes in Computer Science 1563)* (Berlin: Springer) pp 478–87

[2] Cleve R, Ekert A, Macchiavello C and Mosca M 1998 *Proc. R. Soc.* A **454** 339–54
Cleve R, Ekert A, Henderson L, Macchiavello C and Mosca M 1998 *Complexity* **4** 33

[3] Beals R 1997 *STOC: Proc. 29th Ann. ACM Symp. on Theory of Computing* pp 48–53
Høyer P 1997 Efficient quantum transforms *Preprint* quant-ph/9702028
Jozsa R 1998 *Proc. R. Soc.* A **454** 323–37
Kitaev A 1995 Quantum measurements and the Abelian stabilizer problem *Preprint* quant-ph/9511026
Mosca M 1999 Quantum computer algorithms *DPhil Thesis* webpage http://www.qubit.org/people/michele/papers/moscathesis.ps

[4] Mosca M and Ekert A 1999 *Proc. 1st NASA Int. Conf. on Quantum Computing and Quantum Communication (Palm Springs) (Lecture Notes in Computer Science 1509)* (Berlin: Springer) p 174

[5] Beth T and Rötteler M 1998 Polynomial-time solution to the hidden subgroup problem for a class of non-Abelian groups *Preprint* quant-ph/9812070

[6] Brassard G and Høyer P 1997 An exact polynomial time algorithm for Simon's problem *ISTCS'97: Proc. 5th Israeli Symp. on Theory of Computing and Systems* (IEEE Computer Society Press)
Brassard G, Høyer P and Tapp A 1998 *Quantum Counting (Lecture Notes in Computer Science 1443)* (Berlin: Springer) pp 820–31

[7] Grover L 1997 *Phys. Rev. Lett.* **79** 325–8
Jozsa R 1999 Searching in Grover's algorithm *Preprint* quant-ph/9901021