Eigenvalues and Kiteav's Factoring Algorithm.

Stephen McAdam
Department of Mathematics
University of Texas at Austin
mcadam@math.utexas.edu

Introduction. Let U be a unitary operator on a finite dimensional complex inner product space. Suppose $\alpha_1, \alpha_2, \ldots, \alpha_R$ are an unknown set of orthonormal eigenstates of U (with R unknown as well). Since U preserves length, the eigenvalue belonging to α_r has the form $e^{2\pi i c_r}$, for some c_r, with $0 \le c_r < 1$. Suppose the state $\beta = \frac{1}{\sqrt{R}} \sum_{r=1}^{R} \alpha_r$ is known. We will present a process by which a quantum computer can be used so as to have a good chance of giving us enough information that we can approximate one of the c_r to within $1/2^T$. (The computer chooses which c_r; we have no say in the matter.)

We will use U, U^2, U^4, U^8, U^{16}, etc. Therefore, if we want our method to be efficient, we must assume those operators are all efficiently implemented on the computer. (That is the case for Kiteav's U.)

In our situation, each α_r has the coefficient $\frac{1}{\sqrt{R}}$. Suppose instead that $\beta = \sum_{r=1}^{R} a_r \alpha_r$, with varying coefficients a_r. The reader should have no difficulty seeing that our method works equally well in this case, the only difference being that the probability the computer chooses to work with c_r will depend upon the size of $|a_r|^2$, as well as on the other factors we present below. Our work will involve a lot of notation. We felt that having varying coefficients a_r would merely add more notation, without adding more knowledge. Also, Kiteav utilizes the case we will work with.

If we have a known eigenstate α of U, there is a relatively easy way of approximating its eigenvalue. See [G]. Our situation is considerably more complex.

Kiteav, [K], used the ability to find an eigenvector as part of his factoring algorithm. It appears to be fundamentally different from Shor's famous factoring algorithm (see [S] or [M]), although they have points in common.

We assume the reader is familiar with the basics of quantum computation. An introduction can be found in section 2 of [M].

In trying to learn this material, I was unable to find any detailed accounts. (That is one reason I am trying to present it here.) Essentially, I found a few key ideas in the on-line literature. Thinking of them as dots, I have tried to connect them in a coherent and workable fashion. Possibly I have missed some ways of making this more efficient. If so, please forgive me. However, I have deliberately placed more stress on clarity than on efficiency.

1. APPROXIMATIONS.

Suppose $0 \le c < 1$. In this section, we show that if for every integer L with $0 \le L \le T - 3$, we can approximate $\cos 2\pi 2^L c$ to within 1/8, and $\sin 2\pi 2^L c$ to within 1/4, then we can approximate c to within $1/2^T$. (There is one wrinkle, explained below.)

(1.1) Lemma: Suppose x and y are both in the interval $[0, \pi]$ or both in the interval $[\pi, 2\pi]$. If $|\cos x - \cos y| \le 1/4$, then $|x - y| \le \pi/4$.

Proof: We consider the case that $0 \le x \le y \le \pi$, the others being similar. We prove the contrapositive. Suppose $|x - y| > \pi/4$, so that $0 \le x < x + \pi/4 < y \le \pi$. Since the cosine function decreases in the interval $[0, \pi]$, we see that $|\cos x - \cos y| = \cos x - \cos y >$ $\cos x - \cos(x + \pi/4) \ge \min\{\cos z - \cos(z + \pi/4) \mid 0 \le z < 3\pi/4\}$. That minimum occurs when $z = 0$ and equals $1 - \cos \pi/4 > 1/4$, completing the argument.

Notation: Let $\delta > 0$. Then $I[s, \delta]$ will denote the interval $[s - \delta, s + \delta]$.

(1.2) Corollary: Suppose $0 \le C < 1$, and $-1 \le g \le 1$, and $\cos 2\pi C \in I[g, 1/4]$.
If $0 \le C \le 1/2$, then $C \in I[(\cos^{-1} g)/2\pi, 1/8]$.
If $1/2 \le C \le 1$, then $C \in I[1 - (\cos^{-1} g)/2\pi, 1/8]$.

Proof: We do the second case, the first being similar. As the range of the arccosine function is $[0, \pi]$, we see that $2\pi(1 - (\cos^{-1} g)/2\pi) = 2\pi - \cos^{-1} g$ lies between π and 2π, as does $2\pi C$, (since $1/2 \le C \le 1$). The cosines of those two numbers are $\cos 2\pi C$ and g, which by assumption differ by at most 1/4. By (1.1), $|2\pi C - 2\pi(1 - (\cos^{-1} g)/2\pi)| \le \pi/4$. Dividing by 2π gives the result.

Notation: Let $\delta > 0$. $J[0, \delta]$ will denote $[0, \delta] \cup [1 - \delta, 1]$.

TERMINOLOGY: Suppose $0 \le c < 1$. We will say c can be approximated to within δ, if there is a known number s with $c \in I[s, \delta]$. (Specifically, s is an approximation of c to within δ.)
We will say c can be quasi-approximated to within δ if either c can be approximated to within δ, or $c \in J[0, \delta]$ (in the later case, specifically meaning 0 is a quasi-approximation of c to within δ).

EXPLANATION: We explain the use of $J[0, \delta]$ and quasi-approximations. Suppose $\delta > 0$ is very small, and suppose $c \in [1 - \delta, 1]$. Then 0 and c are far apart, but $e^{2\pi i 0}$ is a good approximation to $e^{2\pi i c}$. It would be misleading to say that 0 is an approximation to c. Instead, we note that $c \in J[0, \delta]$, and so we say 0 is a quasi-approximation of c.

(1.3) Lemma: Let $0 \le C < 1$. If $\cos 2\pi C$ can be approximated to within 1/8, and $\sin 2\pi C$ can be approximated to within 1/4, then C can be quasi-approximated to within 1/8.

Proof: By assumption, there is a known g with $\cos 2\pi C \in I[g, 1/8]$. Since $-1 \le \cos 2\pi C \le 1$, if $g < -1$, we can replace it with -1, and if $g > 1$, we can place it with 1. Thus $-1 \le g \le 1$.

First suppose $-1 \le g \le -\sqrt{2}/2 - 1/8$. Then $-1 \le \cos 2\pi C \le (-\sqrt{2}/2 - 1/8) + 1/8 = -\sqrt{2}/2$. If $C \le 1/2$, then $0 \le 2\pi C \le \pi$, and taking arccosines shows $3\pi/4 \le 2\pi C \le \pi$. Thus $3/8 \le C \le 1/2$. On the other hand, if $C > 1/2$, then since $0 < 1 - C < 1/2$ and $\cos 2\pi(1 - C) = \cos 2\pi C$, the preceding shows $3/8 \le 1 - C \le 1/2$, so that $1/2 \le C \le 5/8$. Thus, if $g \le -\sqrt{2}/2 - 1/8$, we know $C \in [3/8, 5/8] = I[1/2, 1/8]$, which is an approximation of C to within 1/8.

Now suppose $\sqrt{2}/2 + 1/8 \le g \le 1$. Then $\sqrt{2}/2 \le \cos 2\pi C \le 1$. If $C \le 1/2$, taking arccosines shows $0 \le 2\pi C \le \pi/4$. Thus $0 \le C \le 1/8$. On the other hand, if $C > 1/2$, we get $0 \le 1 - C \le 1/8$, so that $7/8 \le C \le 1$. Thus, if $g \le \sqrt{2}/2 + 1/8$, we have $C \in [0, 1/8] \cup [7/8, 1] = J[0, 1/8]$, which is a quasi-approximation of C to within 1/8.

For the final case, suppose $-\sqrt{2}/2 - 1/8 < g < \sqrt{2}/2 + 1/8$. Then $(-\sqrt{2}/2 - 1/8) - 1/8 < \cos 2\pi C < (\sqrt{2}/2 + 1/8) + 1/8$. Thus $\cos^2 2\pi C < (\sqrt{2}/2 + 1/4)^2 < .92$, so that $\sin^2 2\pi C > 1 - .92 = .08$. If $\sin 2\pi C$ is positive, it exceeds $\sqrt{.08} > 1/4$. If $\sin 2\pi C$ is negative, it is less than $-1/4$. Since we are assuming $\sin 2\pi C$ can be approximated to within 1/4, we can tell whether it is positive or negative. Therefore, we can tell whether $2\pi C$ is in $(0, \pi)$ or in $(\pi, 2\pi)$, and so whether C is in (0, 1/2) or in (1/2, 1). Suppose the latter, the former case being similar. Since we know $\cos 2\pi C \in I[g, 1/8] \subset I[g, 1/4]$, (1.2) tells us
$C \in I[1 - (\cos^{-1} g)/2\pi, 1/8]$, which is an approximation of C to within 1/8.
(In this final case, we could clearly get a better approximation, but that does not seem to help in what follows.)

(1.4) Remark: In the first two cases of the above proof, we were not able to determine whether $C \le 1/2$, or $C \ge 1/2$. In the final case, we could make that determination. To do that, we needed to know the sign of $\sin 2\pi C$, which we found using our approximation of $\sin 2\pi C$ to within 1/4. That approximation would not be good enough to find the sign, if $\sin 2\pi C$ was too close to 0 (i.e., if $|\sin 2\pi C - 0| \le 1/4 = \sqrt{.08}$). Those "too close" cases were the first two cases of the above proof, handled a different, and less accurate way.

Notation: Let I be an interval, and j and $n > 0$ be numbers. Then $j + I = \{j + x \mid x \in I\}$ and $I/n = \{x/n \mid x \in I\}$.

(1.5) Lemma: Suppose $0 \le c < 1$. Let L be an integer, and let $C = 2^L c - \lfloor 2^L c \rfloor$, (so that C is the fractional part of $2^L c$). Suppose c can be quasi-approximated to within $1/2^{L+2}$, and C can be quasi-approximated to within 1/8. Then c can be quasi-approximated to within $1/2^{L+3}$.

Proof: We know that either $c \in J[0, 1/2^{L+2}]$, or for some known g, $c \in I[g, 1/2^{L+2}]$. Let S denote which ever of $I[g, 1/2^{L+2}]$ or $J[0, 1/2^{L+2}]$ is known to contain c.

We also know that either $C \in J[0, 1/8]$, or for some known h, $C \in I[h, 1/8]$. First suppose $C \in I[h, 1/8]$. Since $0 \le 2^L c < 2^L$, we know $2^L c = j + C$, with j an integer between 0 and $2^L - 1$. Thus $2^L c$ is contained in $\bigcup_{j=0}^{2^L-1} (j + I[h, 1/8])$, and so c is contained in $\bigcup_{j=0}^{2^L-1} (j + I[h, 1/8])/2^L$. However, c is also in S. We claim S only intersects a single $(j + I[h, 1/8])/2^L$ (so it must contain c). (By the phrase "A intersects B", we will mean their intersection is nontrivial.) Suppose the claim is true. Then j can be found. We know $c \in (j + I[h, 1/8])/2^L = I[(j + h)/2^L, 1/2^{L+3}]$, which gives an approximation of c to within $1/2^{L+3}$, as desired. It only remains to prove the claim, which we now do.

We begin with the case $S = I[g, 1/2^{L+2}]$. That interval has length $1/2^{L+1}$. If we can show that the gap between adjacent $(j + I[h, 1/8])/2^L$ exceeds $1/2^{L+1}$, we will be done (in this case). Now the gap between $(j + I[h, 1/8])/2^L$ and $((j + 1) + I[h, 1/8])/2^L$ is $1/2^L$ times the gap between $j + I[h, 1/8]$ and $(j + 1) + I[h, 1/8]$. That last gap is easily seen to be 3/4, and so the gap we want is $3/2^{L+2} > 1/2^{L+1}$, as desired.

We now take the case $S = J[0, 1/2^{L+2}] = [0, 1/2^{L+2}] \cup [1 - 1/2^{L+2}, 1]$. Each of the two intervals in that union has length $1/2^{L+2}$, which we just saw is less than the gap between adjacent $(j + I[h, 1/8])/2^L$. Therefore, if S intersects more than one of the $(j + I[h, 1/8])/2^L$, we must have that $[0, 1/2^{L+2}]$ intersects a unique $(j_1 + I[h, 1/8])/2^L$, and $[1 - 1/2^{L+2}, 1]$ intersects a unique $(j_2 + I[h, 1/8])/2^L$. Translating that last intersection by -1, we see that $[-1/2^{L+2}, 0]$ intersects $-1 + (j_2 + I[h, 1/8])/2^L = (-2^L + j_2 + I[h, 1/8])/2^L$. Therefore, $[-1/2^{L+2}, 1/2^{L+2}]$ (an interval of length $1/2^{L+1}$) intersects $(j + I[h, 1/8])/2^L$ for both $j = j_1$ and $j = -2^L + j_2$. We already know that is impossible, unless $j_1 = -2^L + j_2$. However, all our j satisfy $0 \le j \le 2^L - 1$, so that $j_1 = -2^L + j_2$ is also impossible. Thus, S can only intersect a single $(j + I[h, 1/8])/2^L$, as claimed. This completes the case that $C \in I[h, 1/8]$.

Next, suppose $C \in J[0, 1/8] = [0, 1/8] \cup [7/8, 1]$. We know $2^L c = j + C$ is in $\bigcup_{j=0}^{2^L-1} ([j, j + 1/8] \cup [j + 7/8, j + 1]) =$ $[0, 1/8] \cup [7/8, 9/8] \cup [15/8, 17/8] \cup \cdots \cup [2^L - 1 - 1/8, 2^L - 1 + 1/8] \cup [2^L - 1/8, 2^L]$, and so c is in $[0, 1/8]/2^L \cup [7/8, 9/8]/2^L \cup \cdots \cup [2^L - 1 - 1/8, 2^L - 1 + 1/8]/2^L \cup [2^L - 1/8, 2^L]/2^L$. The intersection of that with S contains c.

Consider the case $S = I[g, 1/2^{L+2}]$. That interval has length $1/2^{L+1}$ while the gap between adjacent intervals in the previous union is $(1/2^L)(3/4) > 1/2^{L+1}$. Thus S can only intersect one of those intervals. If S intersects $[0, 1/8]/2^L = [0, 1/2^{L+3}] = I[1/2^{L+4}, 1/2^{L+4}]$, we have an approximation of c to within $1/2^{L+4}$, better than hoped for. A similar fact holds if S intersects $[2^L - 1/8, 2^L]/2^L = [1 - 1/2^{L+3}, 1]$. On the other hand, if S intersects any of the other intervals in that last union, then we have located c as being in an interval of form $[j/2^L - 1/2^{L+3}, j/2^L + 1/2^{L+3}]$ $= I[j/2^L, 1/2^{L+3}]$, which gives an approximation of c to within $1/2^{L+3}$.

Finally, suppose $S = J[0, 1/2^{L+2}] = [0, 1/2^{L+2}] \cup [1 - 1/2^{L+2}, 1]$. The reader can easily verify that S contains both $[0, 1/8]/2^L$ and $[2^L - 1/8, 2^L]/2^L$, but is disjoint from the other terms in the previous union. Therefore, the intersection of S and that union equals $[0, 1/2^{L+3}] \cup [1 - 1/2^{L+3}, 1] = J[0, 1/2^{L+3}]$. As that contains c, we have a quasi-approximation to within $1/2^{L+3}$.

(1.6) Theorem: Suppose $0 \leq c < 1$. Let $T \geq 3$ be an integer. If for every integer L, with $0 \leq L \leq T - 3$, $\cos(2\pi(2^L c))$ has been approximated within 1/8 and $\sin(2\pi(2^L c))$ has been approximated to within 1/4, then c can be quasi-approximated to within $1/2^T$.

Proof: The case T = 3 is by (1.3). For T > 3, we use induction, assuming c has already been quasi-approximated to within $1/2^{T-1}$. Let L = T – 3. Note that if $C = 2^L c - \lfloor 2^L c \rfloor$, then $\cos 2\pi C = \cos(2\pi(2^L c))$ and $\sin 2\pi C = \sin(2\pi(2^L c))$. The assumptions show that we have an approximation of $2\pi C$ to within 1/8 and an approximation of $\sin 2\pi C$ to within 1/4. By (1.3), we can find a quasi-approximation of C to within 1/8. As our inductive assumption tells us we already have a quasi-approximation of c to within $1/2^{L+2}$, (1.5) shows we can find a quasi-approximation of c to within $1/2^{L+3} = 1/2^T$.

2. THE COMPUTER PROGRAM

Recall that U is a unitary operator, $\alpha_1, \alpha_2, \ldots, \alpha_R$ are unknown orthonormal eigenstates of U, , and the state $\beta = \frac{1}{\sqrt{R}} \sum_{r=1}^{R} \alpha_r$ is known. The eigenvalue belonging to the eigenstate α_r is $\omega_r = e^{2\pi i c_r}$, (some $0 \leq c_r < 1$), and our hope is to find a quasi-approximation of one of those c_r to within $1/2^T$. We have just seen that to do that, it will suffice to approximate $\cos(2\pi(2^L c_r))$ good to within 1/8, and approximate $\sin(2\pi(2^L c_r))$ good to within 1/4, for each L with $0 \leq L \leq T - 3$. We now show how to program the quantum computer so as to have a good chance of accomplishing that, specifically, hoping that the probability of success on any given computer run is at least $1 - 1/2^S$.

We will use auxiliary 'control' qubits. Specifically, we will have sets $S_{c0}, S_{c1}, \ldots S_{c(T-3)}$, each set containing 4H control qubits, and also sets $S_{s0}, S_{s1}, \ldots S_{s(T-3)}$, each set containing H control qubits, where $H \geq 432(T-2)(\log(288(T-2)) + 128S$. We will first use the sets S_{cL}, $(0 \leq L \leq T - 3)$ to approximate cosines, and then use the sets S_{sL} to approximate sines, (the subscripts c and s reminding us what S_{cL} and S_{sL} are for.)

Recall that the Hadamard gate sends the 1-qubit states |0> and |1> to (respectively) $(1/\sqrt{2})(|0> + |1>)$ and $(1/\sqrt{2})(|0> - |1>)$.

We take the first control qubit from S_{c0}, and use it to form the state $|0>\otimes\beta$. We then apply the Hadamard gate to the |0> qubit, resulting in the state $(1/\sqrt{2})(|0> + |1>)\otimes\beta$. We next apply the controlled-U gate. Recall that gate sends $|0>\otimes\beta$ to itself, but sends $|1>\otimes\beta$ to $|1>\otimes U(\beta)$. Thus, our previous state becomes $(1/\sqrt{2})(|0>\otimes\beta + |1>\otimes U(\beta))$. We now apply the Hadamard gate to the first qubit. The result is the state $(1/2)[|0>\otimes\beta + |1>\otimes\beta + |0>\otimes U(\beta) - |1>\otimes U(\beta)]$.

Using that $\beta = (1/\sqrt{R})\sum_{r=1}^{R}\alpha_r$, this equals

$(1/(2\sqrt{R}))[|0\rangle\otimes \sum_{r=1}^{R}\alpha_r + |1\rangle\otimes \sum_{r=1}^{R}\alpha_r + |0\rangle\otimes \sum_{r=1}^{R}U(\alpha_r) - |1\rangle\otimes \sum_{r=1}^{R}U(\alpha_r)]$.

Using that the eigenvalue associated to α_r is ω_r, this equals

$(1/(2\sqrt{R}))[|0\rangle\otimes \sum_{r=1}^{R}\alpha_r + |1\rangle\otimes \sum_{r=1}^{R}\alpha_r + |0\rangle\otimes \sum_{r=1}^{R}\omega_r\alpha_r - |1\rangle\otimes \sum_{r=1}^{R}\omega_r\alpha_r] =$

$(1/\sqrt{R})\sum_{r=1}^{R}\ [(\frac{1+\omega_r}{2})|0\rangle + (\frac{1-\omega_r}{2})|1\rangle] \otimes \alpha_r$.

Letting $\gamma_{0r} = \frac{1+\omega_r}{2}$ and $\rho_{0r} = \frac{1-\omega_r}{2}$, (note: these relate to S_{c0}), then the above state equals

$(1/\sqrt{R})\sum_{r=1}^{R}\ [\gamma_{0r}|0\rangle + \rho_{0r}|1\rangle] \otimes \alpha_r$.

We began with $|0\rangle\otimes\beta = |0\rangle\otimes(1/\sqrt{R})\sum_{r=1}^{R}\ \alpha_r$, and have reached
$(1/\sqrt{R})\sum_{r=1}^{R}\ [\gamma_{0r}|0\rangle + \rho_{0r}|1\rangle] \otimes \alpha_r$ (via the Hadamard gate applied to the first qubit, followed by the controlled-U gate, followed by the Hadamard gate applied to the first qubit).

We now take the second qubit from S_{c0}, and use it to form
$|0\rangle\otimes(1/\sqrt{R})\sum_{r=1}^{R}\ [\gamma_{0r}|0\rangle + \rho_{0r}|1\rangle] \otimes \alpha_r$.

We repeat the previous process, this time using the Hadamard gate applied to the (new) first qubit, followed by the controlled-I⊗U gate, followed by the the Hadamard gate applied to the first qubit. (The controlled-I⊗U gate applied to $|1\rangle \otimes [\gamma_{0r}|0\rangle + \rho_{0r}|1\rangle] \otimes \alpha_r$ gives
$|1\rangle \otimes [\gamma_{0r}|0\rangle + \rho_{0r}|1\rangle] \otimes U(\alpha_r)$.)

At the end of this second round, we will end up at the state

$(1/\sqrt{R})\sum_{r=1}^{R}\ [\gamma_{0r}|0\rangle + \rho_{0r}|1\rangle] \otimes [\gamma_{0r}|0\rangle + \rho_{0r}|1\rangle] \otimes \alpha_r$.

We then utilize the third qubit from S_{c0} to form
$|0\rangle\otimes(1/\sqrt{R})\sum_{r=1}^{R}\ [\gamma_{0r}|0\rangle + \rho_{0r}|1\rangle] \otimes [\gamma_{0r}|0\rangle + \rho_{0r}|1\rangle] \otimes \alpha_r$, and apply the same process, except using the controlled-I⊗I⊗U gate, leading to $(1/\sqrt{R})\sum_{r=1}^{R}\ [\gamma_{0r}|0\rangle + \rho_{0r}|1\rangle]^{\otimes 3} \otimes \alpha_r$.

After we have used the first 4H control qubits in S_{c0} in that way, we reach
$(1/\sqrt{R})\sum_{r=1}^{R}\ [\gamma_{0r}|0\rangle + \rho_{0r}|1\rangle]^{\otimes 4H} \otimes \alpha_r$.

We next start utilizing the second set of 4H control qubits in S_{c1}, and we will replace U with U^2. Thus, we take the first qubit in S_{c1}, and use it to form

$|0\rangle\otimes(1/\sqrt{R})\sum_{r=1}^{R}\ [\gamma_{0r}|0\rangle + \rho_{0r}|1\rangle]^{\otimes 4H}\otimes\alpha_r$. To that, we apply the same process as before, except using the controlled-$I^{\otimes 4H}\otimes(U^2)$ gate, (instead of $I^{\otimes 4H}\otimes(U)$).

Since $U(\alpha_r) = \omega_r\alpha_r$, we see that $U^2(\alpha_r) = \omega_r^2\alpha_r$. Therefore, the process just described leads to

$(1/\sqrt{R})\sum_{r=1}^{R}\ [\gamma_{1r}|0\rangle + \rho_{1r}|1\rangle]\otimes[\gamma_{0r}|0\rangle + \rho_{0r}|1\rangle]^{\otimes 4H}\otimes\alpha_r$, where

$\gamma_{1r} = \dfrac{1+\omega_r^{\,2}}{2}$ and $\rho_{1r} = \dfrac{1-\omega_r^{\,2}}{2}$. (Note: these apply to S_{c1}.)

After the set S_{11} of control qubits has been exhausted, we will be at the state

$(1/\sqrt{R})\sum_{r=1}^{R}\ [\gamma_{1r}|0\rangle + \rho_{1r}|1\rangle]^{\otimes 4H}\otimes[\gamma_{0r}|0\rangle + \rho_{0r}|1\rangle]^{\otimes 4H}\otimes\alpha_r$.

Recall that $0 \le L \le T-3$. For set S_{cL} of control qubits, we use $U^{(2^L)}$, which sends α_r to $\omega_r^{(2^L)}\alpha_r$.

If $\gamma_{Lr} = \dfrac{1+\omega_r^{(2^L)}}{2}$ and $\rho_{Lr} = \dfrac{1-\omega_r^{(2^L)}}{2}$, then we have used all of sets S_{c0}, S_{c1}, …, $S_{c(T-3)}$, (each containing 4H control qubits), we will be at the state

$(1/\sqrt{R})\sum_{r=1}^{R}\ [\gamma_{(T-3)r}|0\rangle + \rho_{(T-3)r}|1\rangle]^{\otimes 4H}\otimes\cdots\otimes[\gamma_{0r}|0\rangle + \rho_{0r}|1\rangle]^{\otimes 4H}\otimes\alpha_r$.

We still have the sets S_{s0}, S_{s1}, …, $S_{s(T-3)}$, (each containing H control qubits), as yet unused. Before using them, it might be helpful to see what we could do if we stopped here. The next lemma begins the illumination, showing how γ_{Lr} is related to $\cos 2\pi 2^L c_r$.

(2.1) Lemma: $|\gamma_{Lr}|^2 = (1/2)(1 + \cos 2\pi 2^L c_r)$ and $|\rho_{Lr}|^2 = (1/2)(1 - \cos 2\pi 2^L c_r)$

Proof: Since $\omega_r = e^{2\pi i c_r}$, we have $\omega_r^{(2^L)} = e^{2\pi i 2^L c_r}$. Therefore,

$\gamma_{Lr} = \dfrac{1+\omega_r^{(2^L)}}{2} = \dfrac{1+e^{2\pi i 2^L c_r}}{2}$, so that one easily sees

$|\gamma_{Lr}|^2 = (1/2)(1 + \cos 2\pi 2^L c_r)$. The case for $|\rho_{Lr}|^2$ is similar.

Staring at an example might be useful. Let 4H = 4 and T = 4 (so $0 \le L \le T-3 = 1$). We have $(1/\sqrt{R})\sum_{r=1}^{R} [\gamma_{1r}|0> + \rho_{1r}|1>]^{\otimes 4} \otimes [\gamma_{0r}|0> + \rho_{0r}|1>]^{\otimes 4} \otimes \alpha_r$.

When expanded, that summation will have a term of form $|0, 0, 1, 0, 0, 1, 1, 0> \otimes \alpha_r$. (Here, the right most bit in |*, *, ...*, *> comes from the first qubit in S_{c0}, while the left most bit comes from the last qubit in S_{c1}) The coefficient of that term will be $(1/\sqrt{R})\gamma_{1r}\gamma_{1r}\rho_{1r}\gamma_{1r}\gamma_{0r}\rho_{0r}\rho_{0r}\gamma_{0r}$. To find the probability of reading (0, 0, 1, 0, 0, 1, 1, 0) from the 8 control bits in $S_{c0} \cup S_{c1}$, we must remember that there are R different choices for α_r. Therefore, the probability of reading (0, 0, 1, 0, 0, 1, 1, 0) is

$$(1/R)\sum_{r=1}^{R} |\gamma_{1r}\gamma_{1r}\rho_{1r}\gamma_{1r}\gamma_{0r}\rho_{0r}\rho_{0r}\gamma_{0r}|^2 = (1/R)\sum_{r=1}^{R} (|\gamma_{1r}|^2)^3 |\rho_{1r}|^2 (|\gamma_{0r}|^2)^2 (|\rho_{0r}|^2)^2 .$$

It is not hard to see that any 8-tuple in which exactly two of the qubits in S_{c0} and one of the qubits in S_{c1} are read as 0, has the same probability (above) of appearing.

If P(1, 2) is the probability of getting a reading in which exactly two of the qubits in S_{c0} and one of the qubits in S_{c1} are read as 0, then P(1, 2) equals the above probability times the number of such 8-tuples. Obviously that number is $\binom{4}{1}\binom{4}{2}$. Therefore,

$$P(1, 2) = \binom{4}{1}\binom{4}{2}(1/R)\sum_{r=1}^{R} (|\gamma_{1r}|^2)^3 |\rho_{1r}|^2 (|\gamma_{0r}|^2)^2 (|\rho_{0r}|^2)^2 .$$

Returning to the general case, the probability that when the control qubits in $S_{c0} \cup S_{c1} \cup \cdots \cup S_{c(T-3)}$ are read, exactly n_L of the qubits in S_{cL} are read as 0 is

$$P(n_{T-3}, n_{T-4}, \ldots n_1, n_0) = \left[\prod_{L=0}^{T-3}\binom{4H}{n_L}\right](1/R)\sum_{r=1}^{R} \prod_{L=0}^{T-3} (|\gamma_{Lr}|^2)^{n_J} (|\rho_{Lr}|^2)^{4H-n_J} .$$

PREVIEW: We claim (and will later show) the above probability is reasonably high only when for some choice of r, each $n_L/4H$ is a good approximation to $|\gamma_{Lr}|^2$. Suppose when we read the control bits, we get exactly n_L zeros from the 4H qubits in S_{cL}, for $0 \le L \le T-3$. Then our claim justifies the hope that for some r, each $|\gamma_{Lr}|^2$ really is approximated by $n_L/4H$. However, by (2.1), $n_L/4H \approx |\gamma_{Lr}|^2 = (1/2)(1 + \cos 2\pi 2^L cr)$, and so $\cos 2\pi 2^L c_r \approx 2n_L/4H - 1$, giving us (we hope) an approximation of $\cos 2\pi 2^L c_r$, for $0 \le L \le T-3$. By choosing H large enough, we will (hopefully) approximate all of those to within 1/8, as required by (1.6).

However, (1.6) also requires approximations of $\sin 2\pi 2^L c_r$ to within 1/4. That is where the sets S_{cL}, $0 \le L \le T-3$, are needed. (Our approximation of $\cos 2\pi 2^L c_r$ can be used to find an approximation of $\sin^2 2\pi 2^L c_r$. That is not good enough. We must approximate $\sin 2\pi 2^L c_r$ itself, since its sign matters. See (1.4).)

As the preview explains, we can now (hopefully) approximate $\cos 2\pi 2^L c_r$ for $0 \leq L \leq T-3$, for some choice of r (chosen by the computer, not by us). However, we also need to approximate $\sin 2\pi 2^L c_r$. To do that, we use the sets S_{cL}, not with U, U^2, U^4, U^8 etc, as earlier, but with -iU, $-iU^2$, $-iU^4$, $-iU^8$, etc.

Since $U^{(2^L)}(\alpha_r) = \omega_r{}^{(2^L)}\alpha$, we see $-iU^{(2^L)}(\alpha_r) = -i\omega_r{}^{(2^L)}\alpha_r$. Now $-i\omega_r{}^{(2^L)} = e^{(3\pi/2)i}(e^{2\pi i c_r})^{(2^L)} = e^{2\pi i(2^L c_r + (3/4))}$. Call that ϖ_{Lr}, so that ϖ_{Lr} is the eigenvalue for the eigenvector α_r of $-iU^{(2^L)}$.

Let $\xi_{Lr} = \dfrac{1+\varpi_{Lr}}{2}$, and $\zeta_{Lr} = \dfrac{1-\varpi_{Lr}}{2}$. (These relate to S_{sL}.)

(2.2) Lemma: $|\xi_{Lr}|^2 = (1/2)(1 + \sin 2\pi 2^L c_r)$ and $|\zeta_{Lr}|^2 = (1/2)(1 - \sin 2\pi 2^L c_r)$.

Proof: $|\xi_{Lr}|^2 = (1/2)(1 + \cos 2\pi(2^L c_r + (3/4)) = (1/2)(1 + \cos(2\pi 2^L c_r + 3\pi/2)) =$

$(1/2)(1 + \sin 2\pi 2^L c_r)$. The other case is similar.

The reader can see that had we done all of our previous work using the various $-iU^{(2^L)}$ instead of the various $U^{(2^L)}$, we would (hopefully) be able to obtain approximations of the various $\sin 2\pi 2^L c_r$.

Of course, we are not allowed to start all over again from our original β, and this time use the $-iU^{(2^L)}$, since that would lead to approximations of $\sin 2\pi 2^L c_s$, with no assurance that $c_s = c_r$. After all, the computer chooses what c it wants, with a new choice on each run.

Instead, we must start with the state

$$(1/\sqrt{R})\sum_{r=1}^{R} [\gamma_{(T-3)r}|0> + \rho_{(T-3)r}|1>]^{\otimes 4H} \otimes \cdots \otimes [\gamma_{0r}|0> + \rho_{0r}|1>]^{\otimes 4H} \otimes \alpha_r,$$

we ended with above. Call the above state β'.

Now take the sets S_{sL} (each containing H control qubits), and apply the previous process, this time using $-iU^{(2^L)}$ for the qubits in S_{sL}.

After completing that, we will be at our truly final state

$(1/\sqrt{R})\sum_{r=1}^{R} \delta_r \otimes \alpha_r,$

where $\delta_r = [\xi_{(T-3)r}|0> + \zeta_{(T-3)r}|1>]^{\otimes H}\otimes\cdots\otimes[\xi_{0r}|0> + \zeta_{0r}|1>]^{\otimes H}\otimes$

$$[\gamma_{(T-3)r}|0> + \rho_{(T-3)r}|1>]^{\otimes 4H}\otimes\cdots\otimes[\gamma_{0r}|0> + \rho_{0r}|1>]^{\otimes 4H}.$$

We now read all of the control bits.

Notation: $P(p_{T-3}, p_{T-4}, \ldots, p_1, p_0, n_{T-3}, n_{T-4}, \ldots, n_1, n_0)$ is the probability that for $0 \le L \le T-3$, when the qubits in S_{cL} are read, they give exactly n_L zeros, and that when the qubits in S_{sL} are read, they give exactly p_L zeros.

To find a formula for the above probability, note that the number of possible readings giving that collection of p_L and n_L is $\left[\prod_{L=0}^{T-3}\binom{H}{p_L}\right]\left[\prod_{L=0}^{T-3}\binom{4H}{n_L}\right]$. All such readings have the same probability of occurring, namely

$$(1/R)\sum_{r=1}^{R}\left[\prod_{L=0}^{T-3}(|\xi_{Lr}|^2)^{p_L}(|\zeta_{Lr}|^2)^{H-p_L}\right]\left[\prod_{L=0}^{T-3}(|\gamma_{Lr}|^2)^{n_L}(|L_{Jr}|^2)^{4H-n_L}\right].$$

Therefore, $P(p_{T-3}, p_{T-4}, \ldots, p_1, p_0, n_{T-3}, n_{T-4}, \ldots, n_1, n_1) =$

$$(1/R)\sum_{r=1}^{R}\left[\prod_{L=0}^{T-3}\binom{H}{p_L}(|\xi_{Lr}|^2)^{p_L}(|\zeta_{Lr}|^2)^{H-p_L}\right]\left[\prod_{L=0}^{T-3}\binom{4H}{n_L}(|\gamma_{Lr}|^2)^{n_L}(|\rho_{Lr}|^2)^{4H-n_L}\right].$$

We will show that if H is large enough, that probability is reasonably high only when for some c_r, each $2p_L/H - 1$ is an approximation of $\sin 2\pi 2^L c_r$ to within 1/4, and each $2n_L/4H - 1$ is an approximation of $\cos 2\pi 2^L c_r$ to within 1/8. Therefore, (1.6) shows we can (within a few computer runs) find a quasi-approximation of c_r to within $1/2^T$. By letting T get large, we can approximate $e^{2\pi i c_r}$, as closely as we wish.

3. PROBABILITIES.

Notation: For integers n and K, with $0 \le n \le K$, let $f_{nK}(x) = x^n(1 - x)^{K-n}$ (for $0 \le x \le 1$).
(In our applications, either K will be H and n will be p_L, or K will be 4H and n will be n_L.)

(3.1) Theorem: $P(p_{T-3}, p_{T-4}, \ldots, p_1, p_0, n_{T-3}, n_{T-4}, \ldots, n_1, n_0) =$

$$(1/R) \sum_{r=1}^{R} \left[\prod_{L=0}^{T-3} \binom{H}{p_L} f_{p_L H}(|\xi_{Lr}|^2)\right]\left[\prod_{L=0}^{T-3} \binom{4H}{n_L} f_{n_L 4H}(|\gamma_{Lr}|^2)\right].$$

Proof: $|\xi_{Lr}|^2 + |\zeta_{Lr}|^2 = 1 = |\gamma_{Lr}|^2 + |\zeta_{Lr}|^2$, and so this is merely a restatement of the previous formula.

We need to study the function $f_{nK}(x)$. We will let $m = K - n$, so that $f_{nK}(x) = x^n(1 - x)^m$. Elementary calculus shows that on our interval, $f_{nK}(x)$ starts at 0 when $x = 0$, rises monotonically to a unique maximum at $x = n/K$, and then falls monotonically back to 0 when $x = 1$.

Notation: M_{nK} will denote $f_{nK}(n/K)$, the maximum value of $f_{nK}(x)$ on [0, 1].

(3.2) Lemma: For $x > 0$, the function $\left(\frac{x+1}{x}\right)^x$ is increasing.

Proof: It will suffice to show its logarithm, $x\log(x + 1) - x\log(x)$, is increasing. It will suffice to show the derivative of that logarithm is positive. That derivative equals $x/(x + 1) - 1 + \log((x + 1)/x) = 1/y - 1 + \log(y)$, where $y = (x + 1)/x$. Now $x > 0$ implies $y > 1$. Let $g(y) = 1/y - 1 + \log(y)$, which we want to exceed 0 when $y > 1$. Suppose $g(y) \le 0$. By the mean value theorem, there is a y_0 with $1 < y_0 < y$, such that $g'(y_0) = (g(y) - g(1))/(y - 1) = (g(y) - 0)/(y - 1) \le 0$. However, $g'(y_0) = (1/y_0)(1 - 1/y_0) > 0$.

(3.3) Lemma: For $0 \le n \le K$, $\binom{K}{n} M_{nK} \le 1$.

Proof: When $n = 0$, $f_{0K}(x) = (1 - x)^K$. The maximum value of that on [0, 1] is $1 = M_{0K}$.
Thus $\binom{K}{0} M_{0K} = 1$.

It is easily seen that $\binom{K}{n}M_{nK} = \binom{K}{K-n}M_{(K-n)K}$, and so it will suffice to show that $\binom{K}{n}M_{nK}$ decreases as n increases from 0 to $\lfloor K/2 \rfloor$. One easily sees that $\binom{K}{n+1}M_{(n+1)K}$ divided by $\binom{K}{n}M_{nK}$ equals $\left(\frac{n+1}{n}\right)^n\left(\frac{m-1}{m}\right)^{m-1}$. We must show that is less than 1 if $0 \le n \le \lfloor K/2 \rfloor - 1$.

Let $s(x) = \left(\frac{x+1}{x}\right)^x$ and $t(y) = \left(\frac{y}{y-1}\right)^{y-1}$. By (3.2), s(x) is increasing for $x > 0$. Also, $t(y) = s(y-1)$. Thus, if $y - 1 > x$, then $t(y) > s(x)$. Let $n \le \lfloor K/2 \rfloor - 1$. Then $m = K - n$ satisfies $m - 1 > n$, and so $t(m) > s(n)$. Thus $\left(\frac{n+1}{n}\right)^n\left(\frac{m-1}{m}\right)^{m-1} =$ $s(n)(t(m))^{-1} < 1$, and we are done.

(3.4) Remark: Let us estimate the minimum value of $\binom{K}{n}M_{nK}$, which the preceding proof shows occurs at $n = \lfloor K/2 \rfloor$. We will assume K is even, so the minimum occurs at $n = K/2$. We leave to the reader the exercise of using Stirling's formula, which says for large R, $R! \sim \sqrt{2\pi R}\left(\frac{R}{e}\right)^R$, to show that when K is large, that minimum is approximately $\sqrt{\frac{2}{\pi K}}$.
(When K = 10, the value is .176..., and the approximation is .178...)

(3.5) Lemma: For $-1 \le x$, $\log(1+x) \le x$. For $-1 \le x \le 0$, $\log(1+x) \le x - x^2/2$.

Proof: Let $g(x) = \log(1+x)$ and $h(x) = x - x^2/2$. Now $g(0) = h(0)$, and $g'(x) - h'(x) = x^2/(1+x) \ge 0$. Those two facts imply the second statement. Now letting $k(x) = x$, we have $g(0) = k(0)$ and $g'(x) - k'(x) = -x/(1+x)$, which is positive exactly when x is negative. The first statement follows.

The following lemma was suggested by R. G. Swan.

(3.6) Lemma: For $0 \le y \le m$, $f_{nK}((n+y)/K) \le M_{nK}/e^{y^2/2m}$.
For $0 \le y \le n$, $f_{nK}((n-y)/K) \le M_{nK}/e^{y^2/2n}$.

Proof: Suppose $0 \le y \le m$. Then $-1 \le -y/m \le 0$. Now $f_{nK}((n+y)/K)/M_{nK} =$

$f_{nK}((n + y)/K)/f_{nK}(n/K) = (1 + y/n)^n(1 - y/m)^m$. The log of that is $n\log(1 + y/n) + m\log(1 - y/m)$, which (3.5) shows is at most $n(y/n) + m(-y/m - y^2/2m^2) = -y^2/2m$. The first statement follows from that.

Now suppose $0 \le y \le n$. Then $\log [f_{nK}((n - y)/K)/M_{nK}] = n\log(1 - y/n) + m\log(1 + y/m)$. By (3.5), that is at most $n(-y/n - y^2/2n^2) + m(y/m) = -y^2/2n$, and the second statement follows.

Remark: Suppose $n > m$. Examples seem to indicate that $f_{nK}(x)$ falls off more rapidly when starting at the maximum M_{nk} at n/K and moving to the right, then when starting at that maximum and moving to the left. The previous lemma hints at that, since when $n > m$, $e^{-y^2/2m} < 3^{-y^2/2n}$.

(3.7) Corollary: If $0 \le (n + y)/K \le 1$, $f_{nK}((n + y)/K) \le M_{nK}/e^{y^2/2K}$.

Proof: This is immediate from (3.6) and the fact that n and m are at most K.

(3.8) Corollary: If $0 \le (n + y)/K \le 1$, $\binom{K}{n} f_{nK}((n + y)/K) \le 1/e^{y^2/2K}$.

Proof: Immediate from (3.7) and (3.3).

Let us recall our goal. We wish to approximate $\cos 2\pi 2^L c_r$ to within 1/8.
Since (2.1) shows $|\gamma_{Lr}|^2 = (1/2)(1 + \cos 2\pi 2^L c_r)$, it will suffice to approximate $|\gamma_{Lr}|^2$ to within 1/16.
Similarly, since we also want to approximate $\sin 2\pi 2^L c_r$ to within 1/4, (2.2) shows we must approximate $|\xi_{Lr}|^2$ to within 1/8.

(3.9) Theorem: Suppose H is large enough that $e^{H/128} > 2^S(4H^2 + 5H + 1)^{T-2}$. (See (3.10).)
Then the probability exceeds $1 - 1/2^S$ that, (after executing the computer program of section 2), reading the control bits will give a reading for which there is some r ($1 \le r \le R$), such that for each L ($0 \le L \le T - 3$) we have $|n_L/4H - |\gamma_{Lr}|^2| \le 1/16$ and $|p_L/H - |\xi_{Lr}|^2| \le 1/8$.
(That is, there is a better than $1 - 1/2^S$ probability that on any given run of the computer program, we will get a reading which allows us to use (1.6) to get a quasi-approximation of some c_r to within $1/2^T$.)

Proof: By (2.1) and (2.2), $|\gamma_{Lr}|^2$ and $|\xi_{Lr}|^2$ are between 0 and 1. Let $y_{Lr}/4H = |\gamma_{Lr}|^2 - n_L/4H$, and $z_{Lr}/H = |\xi_{Lr}|^2 - p_L/H$.

Suppose (supposition A) that for some r and L, we have
$|n_L/4H - |\gamma_{Lr}|^2| = |y_{Lr}/4H| > 1/16$. By (3.8) (with K = 4H), we see $\binom{4H}{n_L} f_{n_L 4H}(|\gamma_{Lr}|^2) =$

$\binom{4H}{n_L} f_{n_L 4H}((n_L + y_{Lr})/4H) \le 1/e^{y_{Lr}^2/8H} < 1/e^{H/128}$ (since $|y_{Lr}| > 4H/16 = H/4$).

Now suppose (supposition B) that $|p_L/H - |\xi_{Lr}|^2| = |z_{Lr}/H| > 1/8$. Then (3.8) (with K = H) similarly shows $\binom{H}{p_L} f_{p_L H}(|\xi_{Lr}|^2) < 1/e^{H/128}$.

By (3.3), each factor of $\left[\prod_{L=0}^{T-3}\binom{H}{p_L} f_{p_L H}(|\xi_{Lr}|^2)\right]\left[\prod_{L=0}^{T-3}\binom{4H}{n_L} f_{n_L 4H}(|\gamma_{Lr}|^2)\right]$ is at most 1.

Thus, if either of suppositions A or B holds for one of those L (and the r appearing in that product), then that product is less than $1/e^{H/128}$ (since that inequality holds for at least one factor). Therefore, if for every r ($1 \le r \le R$), one of supposition A or B holds for some L (depending on r), then

$P(p_{T-3}, p_{T-4}, \ldots, p_1, p_0, n_{T-3}, n_{T-4}, \ldots, n_1, n_0) =$

$$(1/R) \sum_{r=1}^{R} \left[\prod_{L=0}^{T-3}\binom{H}{p_L} f_{p_L H}(|\xi_{Lr}|^2)\right]\left[\prod_{L=0}^{T-3}\binom{4H}{n_L} f_{n_L 4H}(|\gamma_{Lr}|^2)\right] <$$

$(1/R) \sum_{r=1}^{R} (1/e^{H/128}) = (1/R)(R/e^{H/128}) = 1/e^{H/128}$. (Note the vanishing of R. That is why we do not need to know what R is.)

Let # be the number of $(p_{T-3}, p_{T-4}, \ldots, p_1, p_0, n_{T-3}, n_{T-4}, \ldots, n_1, n_0)$ for which one of either supposition A or B holds (for some L depending on r) for each choice of r. Then the probability of getting a reading $(p_{T-3}, p_{T-4}, \ldots, p_1, p_0, n_{T-3}, n_{T-4}, \ldots, n_1, n_0)$ for which one of those two suppositions holds for each r is at most $\#/e^{H/128}$. Therefore, the probability of a reading $(p_{T-3}, p_{T-4}, \ldots, p_1, p_0, n_{T-3}, n_{T-4}, \ldots, n_1, n_0)$ for which there exists an r such that neither of those two suppositions holds is at least $1 - \#/e^{H/128}$. (This latter type of reading is one for which there exists an r with both $|n_L/4H - |\gamma_{Lr}|^2| \le 1/16$ and $|p_L/H - |\xi_{Lr}|^2| \le 1/8$, for all L. Recall, that means $2n_L/4H - 1$ approximates $2\pi 2^L c_r$ to within 1/8, and $2p_L/H - 1$ approximates $\sin 2\pi 2^L c_r$ to within ¼, so that (1.6) can be used.) Therefore, we only need to show, $\#/e^{H/128} < 1/2^S$. Obviously # is at most the total number of possible readings $(p_{T-3}, p_{T-4}, \ldots, p_1, p_0, n_{T-3}, n_{T-4}, \ldots, n_1, n_0)$. Since $0 \le n_L \le 4H$ and $0 \le p_L \le H$, and since there are T – 2 choices for L, we have $\# \le (H + 1)^{T-2}(4H + 1)^{T-2} = (4H^2 + 5H + 1)^{T-2}$. As we want $\#/e^{H/128} < 1/2^S$, it will suffice to show $(4H^2 + 5H + 1)^{T-2}/e^{H/128} < 1/2^S$. However, that is true from the hypotheses.

With H as in the hypothesis of (3.9), we have a probability better than $1 - 1/2^S$ of quasi-approximating some c_r to within $1/2^T$. Recall, the program requires H + 4H = 5H control qubits. H grows roughly as the logs of 2^T and 2^S. We give a sufficient lower bound for H.

(3.10) Lemma: If $T \geq 3$, and if $H \geq 432(T-2)(\log(288(T-2))) + 128S$, then $e^{H/128} > 2^S(4H^2 + 5H + 1)^{T-2}$.

Proof: We first do the case $S = 0$. Our assumption shows H is large enough that $5H^2 \geq 4H^2 + 5H + 1$ and $H \geq 5^4$. Thus $4H^2 + 5H + 1 \leq 5H^2 \leq H^{2.25}$, and so it will suffice to show $e^{H/128} > (H^{2.25})^{T-2}$. Taking logs, we need $H/\log H > 288(T-2)$. For that, $H \geq 432(T-2))(\log(288(T-2)))$ suffices for the case $S = 0$. Now for $S \geq 1$, since $e^{128S/128} > 2^S$, $H \geq 432(T-2))(\log(288(T-2))) + 128S$ suffices.

The reader may wonder why we looked for some not very good approximations to all the $\cos 2\pi 2^L c_r$ and $\sin 2\pi 2^L c_r$, instead of very good approximations to $\cos 2\pi c_r$ and $\sin 2\pi c_r$ (the $L = 0$ cases). The reason is that the latter approach requires a much larger H. Let us see why.

We have quasi-approximated some c_r to $1/2^T$. Thus $2\pi c_r$ can be quasi-approximated to within $\pi/2^{T-1}$, which we call 1/B. That means $\cos 2\pi c_r$ can be approximated to within 1/B. (Outline: we have $2\pi c_r \in I[g, 1/B]$ for a known g. The poorest approximation of $\cos 2\pi c_r$ is when $g = \pm\pi/2$, i.e., when $\cos 2\pi c_r$ is near 0. Now $|\cos 2\pi c_r| \leq |\cos(\pm\pi/2 \pm 1/B)| = |\cos(\pm\pi/2)\cos(\pm 1/B) - \sin(\pm\pi/2)\sin(\pm 1/B)| = \sin(1/B) \leq 1/B$.)

Let us see how big H must be if we tried to approximate $\cos 2\pi c_r$ to within 1/B. To do that, we would only need the set S_{c0} of control qubits from the computer program of section 2. After using all 4H of those qubits, the computer was in state

$$(1/\sqrt{R})\sum_{r=1}^{R} [\gamma_{0r}|0> + \rho_{0r}|1>]^{\otimes 4H} \otimes \alpha_r.$$

The probability of reading exactly n_0 zeros from those 4H control bits is

$$P(n_0) = (1/R)\sum_{r=1}^{R} \binom{4H}{n_0} f_{n_0 4H}{}^{n4H}(|\gamma_{0r}|^2).$$

We hope that for some r, $|\cos 2\pi c_r - (2n_0/4H - 1)| \leq 1/B$. By (2.1), we hope that for some r, We have $||\gamma_{0r}|^2 - n/4H| \leq 1/(2B)$.

We leave to the reader the exercise of mimicking the argument in the proof of (3.9), and show that if $||\gamma_{0r}|^2 - n/4H| > 1/(2B)$ for all r, then $P(n_0) < 1/e^{H/2B^2}$. (Such an n_0 gives failure.) The total number of n_0 is 4H + 1, and so the probability of failure is less than $(4H + 1)/e^{H/2B^2}$. To make that less than1/2 (the case $S = 1$) requires $H > 2B^2 = (2^{2T-1})/\pi^2$, far too large to be efficient.

4. KITEAV'S FACTORING ALGORITHM.

Suppose we wish to factor the large number N. Kiteav's algorithm, begins the same way Shor's algorithm does. We pick a random integer b with $1 < b < N$, and we calculate GCD(b, N). If that GCD is not 1, we have a factor of N. Suppose b is relatively prime to N. Then let R be the order of b mod N in the multiplicative group of units modulo N. It can be shown (see [M, (1.2)] for example) that there is a probability of at least 1/2 that R is even and GCD($b^{R/2} - 1$, N) is a proper factor of N. The only hard part of the above is finding R. Classically, that is very hard. Shor's algorithm gives a quantum computer program that (within a few tries) has a good chance of finding R. Kiteav developed a second approach, which we now explain.

We begin with a set of states $\{|j\rangle \mid 0 \le j \le N-1\}$. (For example, if $N - 1 = 14 < 2^4$, we would need 4 qubits, and the state $|6\rangle$ would represent the actual state $|0, 1, 1, 0\rangle$ of those four qubits, since the binary representation of 6 is 0110.) We work in the complex inner product space having those N states as an orthonormal basis.

Notation: If $0 \le x,y \le N-1$, let $|xy\rangle$ represent the state $|z\rangle$, with $z \equiv xy \bmod N$ and $0 \le z \le N-1$. Similarly, if $1 \le v \le N-1$, with GCD(v, N) = 1, then $|v^{-1}\rangle$ represents $|u\rangle$ where $uv \equiv 1 \bmod N$, and $0 \le u \le N-1$.

Since we are assuming GCD(b, N) = 1, the map U which sends $|j\rangle$ to $|bj\rangle$, gives a permutation of our basis elements, and so can be linearly extended to a unitary operator on our space.

Notation: Let $\omega = e^{2\pi i/R}$. For $1 \le r \le R$, let $\alpha_r = (1/\sqrt{R}) \sum_{t=0}^{R-1} \omega^{tr} |b^{-t}\rangle$.

(4.1) Lemma: $\{\alpha_r \mid 1 \le r \le R\}$ is an orthonormal set of states. Also, α_r is an eigenvector for U, having ω^r for the associated eigenvalue. Finally, $|1\rangle = (1/\sqrt{R}) \sum_{r=1}^{R} \alpha_r$.

Proof: As R is the order of b mod N, the various $|b^{-t}\rangle$, $0 \le t \le R-1$ are distinct vectors in our canonical orthonormal basis. Thus, (using * to denote complex conjugation), the inner product of α_r with α_s (where $1 \le r,s \le R$) is $(1/R) \sum_{t=0}^{R-1} (\omega^{tr})^* (\omega^{ts}) = (1/R) \sum_{t=0}^{R-1} (\omega^{tr})^{-1} (\omega^{ts}) = (1/R) \sum_{t=0}^{R-1} (\omega^{s-r})^t$. If $s = r$, clearly we get 1. If $s \ne r$, using the formula for the sum of a geometric series, we get $\dfrac{(\omega^{s-r})^R - 1}{\omega^{s-r} - 1}$. As ω is a primitive R-th root of unity, that equals 0. That proves the first statement.

Now $U(\alpha_r) = (1/\sqrt{R}) \sum_{t=0}^{R-1} \omega^{tr} U(|b^{-t}\rangle) = (1/\sqrt{R}) \sum_{t=0}^{R-1} \omega^{tr} |b^{-t+1}\rangle =$ $(1/\sqrt{R}) \sum_{t=0}^{R-1} \omega^r \omega^{(t-1)r} |b^{-t+1}\rangle = \omega^r [(1/\sqrt{R}) \sum_{t=0}^{R-1} \omega^{(t-1)r} |b^{-t+1}\rangle]$. Since R is the order of b mod N and $\omega^R = 1$, this equals $\omega^r \alpha_r$, proving the second statement.

Finally, $(1/\sqrt{R})\sum_{r=1}^{R}\alpha_r = (1/R)\sum_{r=1}^{R}\ \sum_{t=0}^{R-1}\omega^{tr}|b^{-t}> =$

$(1/R)\sum_{t=0}^{R-1}(\sum_{r=1}^{R}\omega^{tr})|b^{-t}>$. Now $\sum_{r=1}^{R}\omega^{tr} = \sum_{r=0}^{R-1}\omega^{tr}$ equals 0 for $1 \le t \le R-1$, and equals R when $t = 0$. The final statement follows.

We now have the situation we dealt with in the previous sections; a known state $|1> = \frac{1}{\sqrt{R}}\sum_{r=1}^{R}\alpha_r$ (with R unknown), with the set of α_r comprising an orthonormal set of eigenvectors of the unitary operator U.

The eigenvalue associated with α_r is $\omega^r = e^{2\pi ir/R}$, so the c_r from the previous sections is r/R.

Remark: Our hope is to find some $c_r = r/R$ with GCD(r, R) = 1. If we can do that, then having r/R in reduced terms will allow us to simply read R in the denominator. Therefore, if the process presented below leads to r/R being either 0 or 1, we have failed, and must run the program again. Now we have $1 \le r \le R$. Thus $c_R = R/R = 1$. That breaks our previous rule that $0 \le c_r < 1$. We ignore that transgression, since that case is known to be a failure, and would lead to another try. (Alternately, we could define c_R to be 0.)

We take T large enough that $2^T \ge 2N^2 > 2R^2$. (R is unknown, so N stands in for it. We do know that $R \le \phi(N) < N$.)

Our previous work shows there is probability exceeding $1 - 1/2^S$ that the computer will give us enough information that we can find a quasi-approximation x to some $c_r = r/R$ to within $1/2^T$. The reader can easily verify that if x is 0 or 1, then T is large enough to force r/R to be 0 or 1, both of which are useless to us. That is, if we get x equals 0 or 1, we must run the program again. Therefore, we will assume that we know $r/R \in I[x, 1/2^T]$, where x is known, and $0 < x < 1$. We will now explain how to find r/R.

We remind the reader of part of the study of continued fractions [HW, chapter 10]. For numbers g and $h \neq 0$, we let $[g, h] = g + (1/h)$. Now for integers $a_0, a_1, a_2, \ldots a_k$, all positive except possibly a_0, we let $[a_0, a_1, a_2, \ldots, a_k] = [a_0, [a_1, [a_2, \ldots [a_{m-2}, [a_{m-1}, a_m]]\ldots.]]]$.

Given a real number x, the following algorithm is well known to produce rational numbers that closely approximate x. Let $x_0 = x$ and $a_0 = \lfloor x_0 \rfloor$(the greatest integer equal to or less than x). Inductively, let $x_{i+1} = 1/(x_i - a_i)$ and $a_{i+1} = \lfloor x_{i+1} \rfloor$. (This stops if ever $a_i = x_i$.)
The rational numbers a_0, $[a_0, a_1]$, $[a_0, a_1, a_2]$, $[a_0, a_1, a_2, a_3]$, etc, are known to be successively better approximations to x, and are (vaguely speaking) the best approximations possible using fractions with small denominators. These approximations are called the convergents to x. (They will always appear in reduced terms.)

We remind the reader of a result of Legendre [HW, section 10.15, theorem 184].

(4.2) Theorem: If c and $d > 0$ are integers with $|x - c/d| < 1/(2d^2)$, then c/d is one of the convergents of x.

(4.3) Corollary: If c and d are integers with $0 < d < N$, and if $|x - c/d| \leq 1/(2N^2)$, then c/d is the last term in the sequence of convergents to x having denominator less than N.

Proof: Since $1/(2N^2) < 1/(2d^2)$, (4.2) shows c/d is a convergent to x. Suppose h/k is a later convergent, and $k < N$. By the theory of continued fractions, we have $|x - h/k| < |x - c/d| \leq 1/(2N^2)$. Thus $1/N^2 > |h/k - x| + |x - c/d| \geq |h/k - c/d| = |dh - ck|/dk > |dh - ck|/N^2$. As $|dh - ck|$ is an integer, it must be 0, showing $h/k = c/d$, and contradicting that h/k is a later convergent.

(4.4) Theorem: If $0 < x < 1$ and $r/R \in I[x, 1/2^T]$, then r/R equals the last convergent to x (in the ordered list of those convergents) having denominator less than N.

Proof: We know $|x - r/R| \leq 1/2^T \leq 1/(2N^2)$. Since $R < N$, (4.3) gives the result.

SUMMARY: Run the computer program of section 2. Suppose the reading from it is $(p_{T-3}, p_{T-4}, \ldots, p_1, p_0, n_{T-3}, n_{T-4}, \ldots, n_1, n_0)$. Assume that for some r (with $0 \leq r \leq R - 1$), each p_L/H is an approximation to $\sin 2\pi 2^L(r/R)$ to within 1/4, and each $n_L/4H$ is an approximation of $\cos 2\pi 2^L(r/R)$ to within 1/8. (That assumption will be true with probability at least $1 - 1/2^S$.) Use the arguments in the proofs of (1.3) and (1.4) to find a quasi-approximation x of r/R to within $1/2^T$. If x is 0 or 1, we must run the program again. Suppose $0 < x < 1$. Then use the theory of continued fractions to find the last convergent to x having denominator less than N. That convergent will automatically be in reduced terms, and will equal r/R (which might not be in reduced terms). The denominator of the convergent can be easily tested to determine if it is or is not the order of b mod N. If r/R is in reduced terms, it will be. If it is not (i.e., if $GCD(r, R) \neq 1$), try again. Within a few runs, R should be found.

(4.5) Remarks: a) Recall that (3.10) shows that in the computer program, it will suffice to take H to be the first integer bigger than $432(T - 2)(\log (228(T - 2))) + 128S$. Recall also that we may take T to be the smallest integer with $2^T \geq 2N^2$. Therefore, the number 5H of auxiliary control qubits needed grows slowly, compared to the size of N, the number being factored.

b) If one computes the continued fraction convergents to the number x (in the summary), and lists them as $c_0/d_0, c_1/d_1, c_2/d_2, \ldots$, it can be proven (via a slightly awkward induction) that $c_{i+2} \geq 2c_i$ and $d_{i+2} \geq 2d_i$. Thus $d_{2i} \geq (2^i)d_0 \geq 2^i$. Therefore, we will fairly quickly reach the last convergent with denominator less than N. Also, each such convergent can be calculated quickly.

BIAS: The computer chooses which r/R we get, but we need GCD(r, R) = 1. If an r between 0 and R is chosen at random, there is a good chance that it will be relatively prime to R. However, we must ask if there is a built in bias to the above process that might favor the computer choosing an r not relatively prime to R. That appears to be a somewhat delicate issue. I have seen the statement "all possible r appear with roughly equal probability". I have not seen a proof. We will now consider two aspects of that issue, the first showing that there is some bias, and the second arguing that the bias is somewhat limited.

Recall that M_{nk} is the maximum value of $f_{nK}(x)$ $(0 \le x \le 1)$, which occurs at $x = n/K$.
By (3.1), we see

$P(p_{T-3}, p_{T-4}, \ldots, p_1, p_0, n_{T-3}, n_{T-4}, \ldots, n_1, n_0) \le$

$$(1/R) \sum_{r=1}^{R} \left[\prod_{L=0}^{T-3}\binom{H}{p_L}M_{p_L H}\right]\left[\prod_{L=0}^{T-3}\binom{4H}{n_L}M_{n_L 4H}\right] =$$

$$\left[\prod_{L=0}^{T-3}\binom{H}{p_L}M_{p_L H}\right]\left[\prod_{L=0}^{T-3}\binom{4H}{n_L}M_{n_L 4H}\right].$$

However, the proof of (3.3) combined with (3.4) shows that as n varies from 0 to K, $\binom{K}{n}M_{nK}$ goes from 1, down to approximately $\sqrt{\frac{2}{\pi K}}$ (when n/K is near 1/2), and then climbs back up to 1.

Therefore, if each p_L/H is close to 1/2 and each $n_L/4H$ is close to 1/2, (call that the worst case scenario), then $P(p_{T-3}, p_{T-4}, \ldots, p_1, p_0, n_{T-3}, n_{T-4}, \ldots, n_1, n_0)$ cannot be much bigger than $[(\sqrt{\frac{2}{\pi H}})^{T-3}][(\sqrt{\frac{2}{\pi 4H}})^{T-3}]$. On the other hand, if each p_L/H and $n_L/4H$ is close to either 0 or 1 (call that the best case), that probability has the *potential* of being close to 1 (by the cases n = 0 and n = K in (3.3)). That is the bias. However, we will now show that the bias is mitigated, and that neither the worst nor best case is likely to appear (in particular, the above potential being unfulfilled).

We have shown in (3.9) that we will probably get a reading in which, for each L, we have $|\gamma_{Lr}|^2$ close to $n_L/4H$ and $|\xi_{Lr}|^2$ close to p_L/H. By (2.1) and (2.2) and the fact that $c_r = r/R$, we have $n_L/4H \sim |\gamma_{Lr}|^2 = (1/2)(1 + \cos 2\pi 2^L(r/R))$ and $p_L/H \sim |\xi_{Lr}|^2 = (1/2)(1 + \sin 2\pi 2^L(r/R))$. Therefore, in our actual reading, we will probably have $\cos 2\pi 2^L(r/R)) \sim 2n_L/4H - 1$ and $\sin 2\pi 2^L(r/R)) \sim 2p_L/H - 1$. From that we see that we will probably have $(2p_L/H - 1)^2 + (2n_L/4H - 1)^2 \sim \sin^2 2\pi 2^L(r/R)) + \cos^2 2\pi 2^L(r/R)) = 1$. Therefore, if p_L/H is close to 1/2, then $n_L/4H$ will be close to either 0 or 1, and vice-versa. (This does not fully justify the statement that all r appear with roughly equal probability, but it at least brings it into the realm of the believable.

REFERENCES

[G] Stan Gudder, Quantum Computation, MAA Monthly, v. 110, number 3, March 2003, pp. 181-201].)

[HW] G. H.Hardy and E. M. Wright, An introduction to the theory of numbers (5-th edition), Oxford University Press, New York, 1979.

[K] A. Y. Kiteav, Quantum measurements and the Abelian stabilizer problem, arXiv:quant-ph/9511026v1, November 20, 1995

[M] Stephen McAdam, Shor's algorithms, available at www.ma.utexas.edu/users/mcadam

[S] Peter Shor, Polynomial time algorithms for prime factorization and discrete logarithms on a quantum computer, available at xxx.lanl.gov/abs/quant-ph/9508027