ECE 595: Introduction to Quantum Computing
Final Project: Review of Quantum Phase Estimation

Warren D. (Hoss) Craft

Tues 5/12/2020

Note: References throughout the homework to "Mermin" refer to Mermin's (2007) *Quantum Computer Science: An Introduction* [1], and references to "Nielsen & Chuang" or just "N&C" refer to Nielsen & Chuang's (2010) *Quantum Computation and Quantum Information (10th Anniversary Edition)* [2], and further bracketed citations are omitted.

(1) Intro

Nielsen & Chuang (2010) segue from the quantum Fourier transform to quantum phase estimation (QPE) by remarking that "The Fourier transform is the key to a general procedure known as phase estimation, which in turn is the key for many quantum algorithms" (pg 221). Among the many quantum algorithms for which QPE is key are the quantum order-finding and quantum factoring algorithms, covered in N&C §§5.31–5.32 (pp 226–234).

QPE Problem Statement (based on N&C, and the QPE page in *Wikipedia* [3]):

> Let U be a unitary operator that operates on the m-Qbit $|u\rangle$ such that $U|u\rangle = e^{2\pi i\varphi}|u\rangle$ — *i.e.*, $|u\rangle$ is an eigenstate of the operator U with eigenvalue $e^{2\pi i\varphi}$, where φ is an unknown real value with $0 \le \varphi < 1$.
>
> Goal: estimate φ (which is then equivalent to estimating the eigenvalue $e^{2\pi i\varphi}$).

The eigenvalue $e^{2\pi i\varphi}$

Although the eigenvalue form $e^{2\pi i\varphi}$ at first seems highly specific or idiosyncratic, as pointed out in [3], the eigenvalue λ_u associated with eigenstate $|u\rangle$ can always be written in the form $e^{2\pi i\varphi}$ because U is a unitary operator over a complex vector space, so its eigenvalues must be expressible as complex numbers with modulus 1.

More explicitly, suppose $\lambda_u = a + bi$ for real values a and b. Then $\theta = 2\pi\varphi = \tan^{-1}(\frac{b}{a})$ gives $\varphi = \frac{1}{2\pi}\tan^{-1}(\frac{b}{a})$ (with suitable accommodations for cases where $a = 0$ and using the signs of a and b to decide on the correct quadrant for the inverse tangent result) to produce $\lambda_u = a + bi = \cos(2\pi\varphi) + i\sin(2\pi\varphi) = e^{2\pi i\varphi}$.

(2) The QPE Procedure: Stage 1

(a) Registers and Initial Conditions

The QPE algorithm utilizes 2 registers. The 1st register consists of t Qbits, each initially in the state $|0\rangle$. The 2nd register consists of m Qbits and is used to store the eigenstate $|u\rangle$. The circuit for the algorithm is shown in Figure (1) below, which is a reasonable facsimile of N&C's Figure 5.2 (pg 222), with Qbits going from top to bottom in order of decreasing significance — *e.g.*, we"' refer to the upper-most Qbit in the 1st register at Qbit $t-1$ and the bottom-most Qbit in the 1st register as Qbit 0.

(b) Hadamard applied to 1st register

The QPE algorithm begins with what Mermin describes as the "standard trick": the application of a Hadamard to each Qbit of the 1st register to produce the uniformly-weighted superposition:

$$H_{t-1}H_{t-2}\dots H_1H_0\,|00\dots00\rangle = \frac{1}{\sqrt{2}}(|0\rangle+|1\rangle)\frac{1}{\sqrt{2}}(|0\rangle+|1\rangle)\dots\frac{1}{\sqrt{2}}(|0\rangle+|1\rangle)\frac{1}{\sqrt{2}}(|0\rangle+|1\rangle) \tag{1}$$

$$= \frac{1}{2^{t/2}}\sum_{x=0}^{2^t-1}|x\rangle_t \tag{2}$$

(c) Controlled U^{2^j} operations applied to the 2nd register ...

The QPE algorithm continues with the successive controlled U^{2^j} operations applied to the 2nd register, with each U^{2^j} application being controlled by the jth Qbit in the 1st register.

To understand a bit better the eventual result in this first stage, shown along the right-hand side of Figure (1), consider first the process experienced by just the 0th Qbit in the first register:

$$
\begin{aligned}
C^{U^{2^0}}_{0|u\rangle} H_0 |0\rangle |u\rangle &= \frac{1}{\sqrt{2}} C^{U^{2^0}}_{0|u\rangle} (|0\rangle + |1\rangle) |u\rangle && (3)\\
&= \frac{1}{\sqrt{2}} C^{U^{2^0}}_{0|u\rangle} (|0\rangle |u\rangle + |1\rangle |u\rangle) && (4)\\
&= \frac{1}{\sqrt{2}} (|0\rangle |u\rangle + |1\rangle U^{2^0} |u\rangle) && (5)\\
&= \frac{1}{\sqrt{2}} (|0\rangle |u\rangle + |1\rangle e^{2\pi i(2^0\varphi)} |u\rangle) && (6)\\
&= \frac{1}{\sqrt{2}} (|0\rangle + e^{2\pi i(2^0\varphi)} |1\rangle) |u\rangle && (7)\\
&= |\psi_0\rangle |u\rangle && (8)
\end{aligned}
$$

where $|\psi_0\rangle$ is the result for Qbit 0 in the 1st register, shown along the right-hand side of Figure (1).
Then consider the next part of the process for Qbit 1:

$$
\begin{aligned}
C^{U^{2^1}}_{1|u\rangle} H_1 |0\rangle |\psi_0\rangle |u\rangle &= \frac{1}{\sqrt{2}} C^{U^{2^1}}_{1|u\rangle} (|0\rangle + |1\rangle) |\psi_0\rangle |u\rangle && (9)\\
&= \frac{1}{\sqrt{2}} (|0\rangle |\psi_0\rangle |u\rangle + |1\rangle |\psi_0\rangle U^{2^1} |u\rangle) && (10)\\
&= \frac{1}{\sqrt{2}} (|0\rangle |\psi_0\rangle |u\rangle + |1\rangle |\psi_0\rangle e^{2\pi i(2^1\varphi)} |u\rangle) && (11)\\
&= \frac{1}{\sqrt{2}} (|0\rangle + e^{2\pi i(2^1\varphi)} |1\rangle) |\psi_0\rangle |u\rangle && (12)\\
&= |\psi_1\rangle |\psi_0\rangle |u\rangle && (13)
\end{aligned}
$$

Eventually, of course, this continued process leads to the 1st-stage result of $|\psi_{t-1}\rangle \dots |\psi_1\rangle |\psi_0\rangle |u\rangle$.

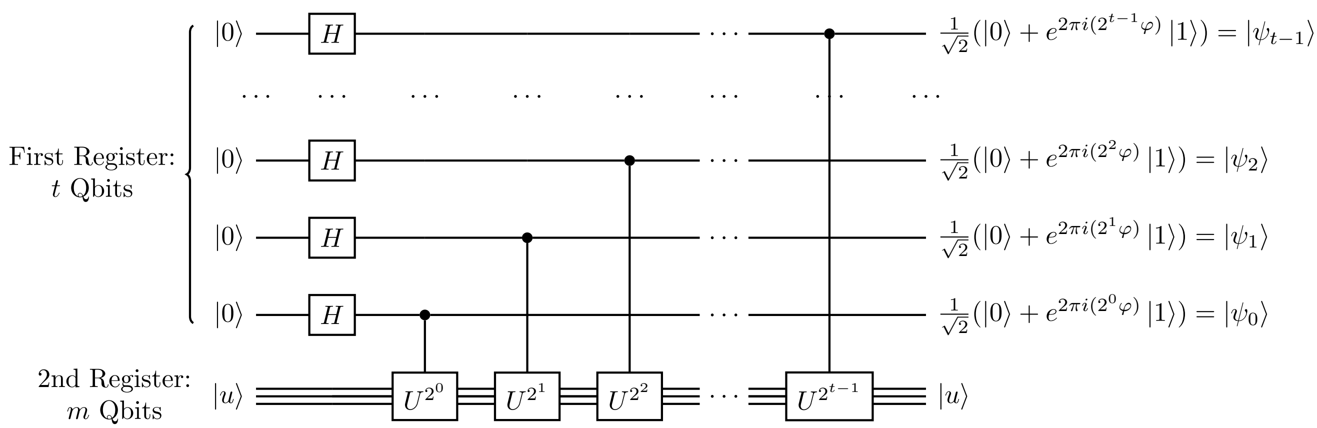

Figure 1: The quantum circuit diagram for what Nielsen & Chuang (2010) call the "first stage" of the quantum phase estimation algorithm (largely a replication of their Figure 5.2, pg 222).

(d) Final state of 1st register at the end of 1st stage ......

The final state of the 1st register at the end of this 1st stage is then:

$$\frac{1}{\sqrt{2}}\left(|0\rangle + e^{2\pi i 2^{t-1}\varphi}|1\rangle\right) \otimes \frac{1}{\sqrt{2}}\left(|0\rangle + e^{2\pi i 2^{t-2}\varphi}|1\rangle\right) \otimes \cdots \otimes \frac{1}{\sqrt{2}}\left(|0\rangle + e^{2\pi i 2^{1}\varphi}|1\rangle\right) \otimes \frac{1}{\sqrt{2}}\left(|0\rangle + e^{2\pi i 2^{0}\varphi}|1\rangle\right) \tag{14}$$

$$= \frac{1}{2^{t/2}}\left(|0\rangle_t + e^{2\pi i\varphi}|1\rangle_t + e^{2\pi i(2\varphi)}|2\rangle_t + \cdots + e^{2\pi i((2^t-1)\varphi)}\left|2^t-1\right\rangle_t\right) \tag{15}$$

$$= \frac{1}{2^{t/2}}\sum_{k=0}^{2^t-1} e^{2\pi i k\varphi}|k\rangle_t \tag{16}$$

where the subscript t is a reminder we are working over a t-Qbit register.

(3) Intercalary: Nielsen & Chuang's Exercise 5.7 (pg 222):

> **Exercise 5.7:** Additional insight into the circuit in Figure 5.2 may be obtained by showing, as you should now do, that the effect of the sequence of controlled-U operations like that in Figure 5.2 is to take the state $|j\rangle|u\rangle$ to $|j\rangle U^j|u\rangle$. (Note that this does not depend on $|u\rangle$ being an eigenstate of U.)

Solution. We take $|j\rangle = |j\rangle_t$ — *i.e.*, j is an integer value expressible in t bits, so that

$$j = \sum_{i=0}^{t-1}\alpha_i 2^i, \text{ for } \alpha_i \in \{0,1\} \tag{17}$$

and thus

$$|j\rangle = |\alpha_{t-1}\rangle|\alpha_{t-1}\rangle\ldots|\alpha_1\rangle|\alpha_0\rangle \tag{18}$$

Then the effect of the sequence of controlled-U operations like that in Figure 5.2 is to effect the transformation:

$$|j\rangle|u\rangle \mapsto |j\rangle\prod_{i=0}^{t-1} e^{2\pi i(\alpha_i 2^i\varphi)}|u\rangle \tag{19}$$

because for each non-zero α_i in the expansion of j, we will obtain a (non-unitary) factor of $e^{2\pi i(2^i\varphi)}$. The product of the exponential functions can be re-expressed as a sum in the exponent, where the sum is exactly our formula for j:

$$|j\rangle\prod_{i=0}^{t-1} e^{2\pi i(\alpha_i 2^i\varphi)}|u\rangle = |j\rangle e^{2\pi i(\sum_{i=0}^{t-1}(\alpha_i 2^i)\varphi)}|u\rangle \tag{20}$$

$$= |j\rangle e^{2\pi i(j\varphi)}|u\rangle \tag{21}$$

$$= |j\rangle\left(e^{2\pi i\varphi}\right)^j|u\rangle \tag{22}$$

$$= |j\rangle U^j|u\rangle \tag{23}$$

$$\tag{24}$$

where in the end we used the fact that $U|u\rangle = e^{2\pi i\varphi}|u\rangle$ and thus $U^j|u\rangle = (e^{2\pi i\varphi})^j|u\rangle$.

Some simple concrete examples of this result are shown worked out from scratch in Table 1 for j defined on $t=3$ Qbits.

(4) 2nd Stage: Applying the Inverse Fourier Transform (and N&C's Eqs 5.21–5.22)

Suppose for the moment that our phase φ can be represented exactly with our t bits in the first register and so we can express our phase in binary decimal form as $\varphi = 0.\varphi_1\varphi_2\ldots\varphi_{t-1}\varphi_t$, where each $\varphi_i \in \{0,1\}$ — that is, the phase φ can be expressed as:

$\lvert j\rangle \lvert u\rangle$ input	output	simplified output
$\lvert 001\rangle \lvert u\rangle$	$\lvert 001\rangle U^{2^0} \lvert u\rangle$	$\lvert 001\rangle U^1 \lvert u\rangle = \lvert 1\rangle_3 U^1 \lvert u\rangle$
$\lvert 010\rangle \lvert u\rangle$	$\lvert 010\rangle U^{2^1} \lvert u\rangle$	$\lvert 010\rangle U^2 \lvert u\rangle = \lvert 2\rangle_3 U^2 \lvert u\rangle$
$\lvert 011\rangle \lvert u\rangle$	$\lvert 011\rangle U^{2^1} U^{2^0} \lvert u\rangle$	$\lvert 011\rangle U^2 U^1 \lvert u\rangle = \lvert 3\rangle_3 U^3 \lvert u\rangle$
$\lvert 100\rangle \lvert u\rangle$	$\lvert 100\rangle U^{2^2} \lvert u\rangle$	$\lvert 100\rangle U^4 \lvert u\rangle = \lvert 4\rangle_3 U^4 \lvert u\rangle$
$\lvert 101\rangle \lvert u\rangle$	$\lvert 101\rangle U^{2^2} U^{2^0} \lvert u\rangle$	$\lvert 101\rangle U^4 U^1 \lvert u\rangle = \lvert 5\rangle_3 U^5 \lvert u\rangle$
$\lvert 110\rangle \lvert u\rangle$	$\lvert 110\rangle U^{2^2} U^{2^1} \lvert u\rangle$	$\lvert 110\rangle U^4 U^2 \lvert u\rangle = \lvert 6\rangle_3 U^6 \lvert u\rangle$

Table 1: Examples of results for Exercise 5.7, with j defined on 3 Qbits.

$$\varphi = \frac{\varphi_1}{2^1} + \frac{\varphi_2}{2^2} + \cdots + \frac{\varphi_{t-1}}{2^{t-1}} + \frac{\varphi_t}{2^t} \tag{25}$$

$$= \sum_{j=1}^{t} \varphi_j 2^{-j} \tag{26}$$

Then, as shown in N&C's Eq 5.21 (pg 22), the first register state in expression (14) resulting from the first stage of processing can be re-expressed as:

$$\begin{aligned} &\frac{1}{2^{t/2}} \left(\lvert 0\rangle + e^{2\pi i 2^{t-1}(0.\varphi_1\varphi_2\ldots\varphi_t)} \lvert 1\rangle\right) \left(\lvert 0\rangle + e^{2\pi i 2^{t-2}(0.\varphi_1\varphi_2\ldots\varphi_t)} \lvert 1\rangle\right) \cdots \\ &\qquad \left(\lvert 0\rangle + e^{2\pi i 2^{1}(0.\varphi_1\varphi_2\ldots\varphi_t)} \lvert 1\rangle\right) \left(\lvert 0\rangle + e^{2\pi i 2^{0}(0.\varphi_1\varphi_2\ldots\varphi_t)} \lvert 1\rangle\right) \end{aligned} \tag{27}$$

$$\begin{aligned} = &\frac{1}{2^{t/2}} \left(\lvert 0\rangle + e^{2\pi i(\varphi_1\varphi_2\ldots\varphi_{t-1}.\varphi_t)} \lvert 1\rangle\right) \left(\lvert 0\rangle + e^{2\pi i(\varphi_1\varphi_2\ldots.\varphi_{t-1}\varphi_t)} \lvert 1\rangle\right) \cdots \\ &\qquad \left(\lvert 0\rangle + e^{2\pi i(\varphi_1.\varphi_2\ldots\varphi_t)} \lvert 1\rangle\right) \left(\lvert 0\rangle + e^{2\pi i(0.\varphi_1\varphi_2\ldots\varphi_t)} \lvert 1\rangle\right) \end{aligned} \tag{28}$$

$$= \frac{1}{2^{t/2}} \left(\lvert 0\rangle + e^{2\pi i(0.\varphi_t)} \lvert 1\rangle\right) \left(\lvert 0\rangle + e^{2\pi i(0.\varphi_{t-1}\varphi_t)} \lvert 1\rangle\right) \cdots \left(\lvert 0\rangle + e^{2\pi i(0.\varphi_2\ldots\varphi_t)} \lvert 1\rangle\right) \left(\lvert 0\rangle + e^{2\pi i(0.\varphi_1\varphi_2\ldots\varphi_t)} \lvert 1\rangle\right) \tag{29}$$

where the leading digits are gradually "dropped" from the expression for φ as we apply successive multiplications by 2. Since it may not be obvious how/why we can simply drop those leading φ_j digits with each successive multiplication by a factor of 2 in the exponential function, it is worth considering the step from the expression in (28) to expression (29), for the exponential function in just the second-to-last right-hand term:

$$e^{2\pi i 2^1(0.\varphi_1\varphi_2\ldots\varphi_t)} = e^{2\pi i(\varphi_1.\varphi_2\ldots\varphi_t)} \tag{30}$$

$$= e^{2\pi i(\varphi_1 + 0.\varphi_2\ldots\varphi_t)} \tag{31}$$

$$= e^{2\pi i(\varphi_1)} e^{2\pi i(0.\varphi_2\ldots\varphi_t)} \tag{32}$$

$$= (1) e^{2\pi i(0.\varphi_2\ldots\varphi_t)} \tag{33}$$

$$= e^{2\pi i(0.\varphi_2\ldots\varphi_t)} \tag{34}$$

In other words, since $e^{2\pi i \varphi_j} = 1$ whenever $\varphi \in \{0, 1\}$, those leading digits φ_j each time just produce another factor of 1 each time and can be removed.

What's beautiful about the resulting expression in (29), though, is that it matches the quantum Fourier transform output shown in N&C's Eq (5.4) (pg 218):

$$\lvert j_1, j_2, \ldots, j_n\rangle \xrightarrow{QFT} \frac{1}{2^{t/2}} \left(\lvert 0\rangle + e^{2\pi i(0.j_n)} \lvert 1\rangle\right) \left(\lvert 0\rangle + e^{2\pi i(0.j_{n-1}j_n)} \lvert 1\rangle\right) \cdots \left(\lvert 0\rangle + e^{2\pi i(0.j_1 j_2 \cdots j_n)} \lvert 1\rangle\right) \tag{35}$$

Thus applying the inverse quantum Fourier transform to our 1st-stage 1st-register result gives us:

$$\frac{1}{2^{t/2}}\left(|0\rangle + e^{2\pi i(0.\varphi_t)}|1\rangle\right)\left(|0\rangle + e^{2\pi i(0.\varphi_{t-1}\varphi_t)}|1\rangle\right)\cdots\left(|0\rangle + e^{2\pi i(0.\varphi_1\varphi_2\ldots\varphi_t)}|1\rangle\right) \xrightarrow{QFT^{-1}} |\varphi_1\varphi_2\ldots\varphi_t\rangle \tag{36}$$

and thus (as Nielsen & Chuang point out at the bottom of pg 222), a measurement in the computational basis gives us φ exactly (when φ could be expressed exactly in the t or fewer Qbits available in the first register).

(5) 2nd Stage: Applying the Inverse Fourier Transform when φ can only be approximated in the available t Qbits

Here we diverge a bit from Nielsen & Chuang's presentation and take our cue from the presentation in the *Wikipedia* presention on this same topic [3].

Recall that the state of the first register at the end of Stage 1 (just before the inverse Fourier transform) is given by:

$$\frac{1}{2^{t/2}}\sum_{k=0}^{2^t-1} e^{2\pi i k\varphi}|k\rangle_t \tag{37}$$

Applying the inverse Fourier transform then gives:

$$\begin{aligned}
\mathcal{F}^{-1}\left\{\frac{1}{2^{t/2}}\sum_{k=0}^{2^t-1} e^{2\pi i k\varphi}|k\rangle_t\right\} &= \frac{1}{2^{t/2}}\sum_{k=0}^{2^t-1} e^{2\pi i k\varphi}|k\rangle_t \frac{1}{2^{t/2}}\sum_{x=0}^{2^t-1} e^{\frac{-2\pi i k x}{2^t}}|x\rangle & (38)\\
&= \frac{1}{2^t}\sum_{x=0}^{2^t-1}\sum_{k=0}^{2^t-1} e^{2\pi i k\varphi}|k\rangle_t e^{\frac{-2\pi i k x}{2^t}}|x\rangle \quad \text{[combine coeffs and rearrange summations]} & (39)\\
&= \frac{1}{2^t}\sum_{x=0}^{2^t-1}\sum_{k=0}^{2^t-1} e^{\frac{-2\pi i k}{2^t}(x-2^t\varphi)}|x\rangle \quad \text{[combine exponents]} & (40)
\end{aligned}$$

Recall now that $\varphi = 0.\varphi_1\varphi_2\varphi_3\ldots\varphi_{t-1}\varphi_t\varphi_{t+1}\ldots \in [0,1]$ (because we now require more than t bits to represent φ). We could thus approximate φ by rounding to the nearest integer the value $2^t\varphi = \varphi_1\varphi_2\varphi_3\ldots\varphi_{t-1}\varphi_t.\varphi_{t+1}\ldots$, and using the resulting value $\tilde{\varphi} = \frac{\varphi_1'\varphi_2'\varphi_3'\ldots\varphi_{t-1}'\varphi_t'}{2^t}$ as our estimate for φ. So we take $2^t\varphi = a + 2^t\delta$, where a is the nearest integer to $2^t\varphi$ and $0 \le |2^t\delta| \le \frac{1}{2}$.

We can then re-express the post-inverse Fourier transform state of the first register as (see the "Phase approximation representation" section of [3]):

$$\begin{aligned}
\frac{1}{2^t}\sum_{x=0}^{2^t-1}\sum_{k=0}^{2^t-1} e^{\frac{-2\pi i k}{2^t}(x-2^t\varphi)}|x\rangle &= \frac{1}{2^t}\sum_{x=0}^{2^t-1}\sum_{k=0}^{2^t-1} e^{\frac{-2\pi i k}{2^t}(x-(a+2^t\delta))}|x\rangle & (41)\\
&= \frac{1}{2^t}\sum_{x=0}^{2^t-1}\sum_{k=0}^{2^t-1} e^{\frac{-2\pi i k}{2^t}((x-a)-2^t\delta)}|x\rangle & (42)\\
&= \frac{1}{2^t}\sum_{x=0}^{2^t-1}\sum_{k=0}^{2^t-1} e^{\frac{-2\pi i k}{2^t}(x-a)}e^{2\pi i k\delta}|x\rangle & (43)\\
&= A & (44)
\end{aligned}$$

(6) 3rd Stage: Measuring First Register in Computational Basis (see the "Measurement" section in [3])

When we now perform a measurement on the first register in the computational basis, we obtain the integer value a

with probability given by:

$$\begin{aligned} p(a) &= |\langle a|A\rangle|^2 && (45)\\ &= \left| \langle a| \frac{1}{2^t} \sum_{x=0}^{2^t-1} \sum_{k=0}^{2^t-1} e^{\frac{-2\pi i k}{2^t}(x-a)} e^{2\pi i k \delta} |x\rangle \right|^2 && (46)\\ &= (\frac{1}{2^t})^2 \left| \sum_{x=0}^{2^t-1} \sum_{k=0}^{2^t-1} e^{\frac{-2\pi i k}{2^t}(x-a)} e^{2\pi i k \delta} \langle a|x\rangle \right|^2 && (47) \end{aligned}$$

and there when summing over x, the only non-zero term occurs when $x = a$ (when $\langle a|x\rangle = \langle a|a\rangle = 1$, otherwise $\langle a|x\rangle = 0$). Thus we have:

$$p(a) = \frac{1}{2^{2t}} \left| \sum_{k=0}^{2^t-1} e^{2\pi i k \delta} \right|^2 \tag{48}$$

Notice that when $\delta = 0$ (and φ can be exactly represented with the t bits in the first register), we get:

$$p(a)|_{\delta=0} = \frac{1}{2^{2t}} \left| \sum_{k=0}^{2^t-1} 1 \right|^2 = \frac{1}{2^{2t}} (2^t)^2 = 1 \tag{49}$$

Thus when $2^t\varphi = a$, we obtain the exact value for $2^t\varphi$ with the first register giving $|\varphi_1\varphi_2 \ldots \varphi_t\rangle$ and measured as such with probability 1, consistent with our earlier result when we assumed that φ could be represented exactly within the t bits of the first register.

If $\delta \neq 0$, the summation term can be interpreted as a truncated geometric series of the form $\sum_{j=0}^{N} cr^k$, with constant coefficient $c = 1$ and ratio $r = e^{2\pi i \delta}$, and thus:

$$\sum_{k=0}^{2^t-1} e^{2\pi i k \delta} = \sum_{k=0}^{2^t-1} (e^{2\pi i \delta})^k = \frac{1 - (e^{2\pi i \delta})^{2t}}{1 - (e^{2\pi i \delta})} = \frac{1 - e^{2\pi i 2^t \delta}}{1 - e^{2\pi i \delta}} \tag{50}$$

Plugging back into our formula for $p(a)$, then, for $\delta \neq 0$ we have:

$$p(a)|_{\delta\neq 0} = \frac{1}{2^{2t}} \left| \frac{1 - e^{2\pi i 2^t \delta}}{1 - e^{2\pi i \delta}} \right|^2 \tag{51}$$

Interestingly, we can give a lower bound on this probability for obtaining such an estimate for φ, as developed at the end of [3]:

$$p(a)|_{\delta\neq 0} = \frac{1}{2^{2t}} \left| \frac{1 - e^{2\pi i 2^t \delta}}{1 - e^{2\pi i \delta}} \right|^2 = \frac{1}{2^{2t}} \left| \frac{2\sin(\pi 2^t \delta)}{2\sin(\pi\delta)} \right|^2 \tag{52}$$

because

$$\begin{aligned} |1 - e^{2ix}|^2 &= |1 - (\cos(2x) + i\sin(2x))|^2 && (53)\\ &= |(1 - \cos(2x)) - i\sin(2x)|^2 && (54)\\ &= |2\sin^2(x) - i(2\sin(x)\cos(x))|^2 && (55)\\ &= 4\sin^4(x) + 4\sin^2(x)\cos^2(x) && (56)\\ &= 4\sin^2(x)\left(\sin^2(x) + \cos^2(x)\right) && (57)\\ &= 4\sin^2(x) = |2\sin(x)|^2 && (58) \end{aligned}$$

Then:

$$
\begin{aligned}
p(a)|_{\delta\neq 0} &= \frac{1}{2^{2t}}\frac{|2\sin(\pi 2^t\delta)|^2}{|2\sin(\pi\delta)|^2} && && (59)\\
&\geq \frac{1}{2^{2t}}\frac{|\sin(\pi 2^t\delta)|^2}{|\pi\delta|^2} && \text{because } |\sin(\pi\delta)| \leq |\pi\delta| \text{ for small } \delta && (60)\\
&\geq \frac{1}{2^{2t}}\frac{|2\cdot 2^t\delta|^2}{|\pi\delta|^2} && \text{because } |2\cdot 2^t\delta| \leq |\sin\bigl(|2\cdot 2^t\delta|\bigr)| \text{ for } \delta \leq \frac{1}{2^{t+1}} && (61)\\
&= \frac{4}{\pi^2} \approx 0.4053 && && (62)
\end{aligned}
$$

Thus we have over a 40% probability of measuring the best t-bit estimate of $2^n\varphi$ (and thus φ) when φ cannot be represented exactly in the t Qbits of the first register.

References

[1] N. David Mermin. *Quantum Computer Science: An Introduction*. New York, NY: Cambridge University Press, 2007.

[2] Michael A. Nielsen and Isaac L. Chuang. *Quantum Computation and Quantum Information (10th Anniversary Edition)*. New York, NY: Cambridge University Press, 2010.

[3] Wikipedia contributors. *Quantum phase estimation algorithm — Wikipedia, The Free Encyclopedia*. `https://en.wikipedia.org/w/index.php?title=Quantum_phase_estimation_algorithm&oldid=953506229`. [Online; last accessed 9-May-2020]. 2020.