Lecture 15: Electron-phonon coupling on quantum computers
Reading: Macridin et al PRA (2019).

1 Introduction

Throughout the class we have focused on quantum simulation on fermionic systems. What about bosonic systems, and fermionic-bosonic interactions? We can certainly map bosonic systems to qubits although the mapping is more complicated because the occupation of bosons is no longer limited to 0 or 1 particles. Further, remember that qubit operators are neither fermionic nor bosonic, and so we will need an operator mapping to obtain a truly bosonic representation in qubits.

There have been a number of ideas presented for how to overcome these problems through the years. One idea is in Ref. [1]. Let a bosonic operator B be given as:

$$B = \frac{1}{\sqrt{N}} \sum_{i=1}^{N} a_i \tag{1.1}$$

where a_i are qubit operators that obey the "in-between" commutation relations $\{a_i, a_i^\dagger\} = 1$, $[a_i, a_j^\dagger] = 0$ for $i \neq j$. Then the commutator of B is:

$$[B, B^\dagger] = \frac{1}{N} \sum_{ij} [a_i, a_j^\dagger] = \frac{1}{N} \sum_i (1 - 2n_i) = 1 - \frac{2n}{N} \tag{1.2}$$

where n is the occupied qubit number and N is the number of qubits. If $n \ll N$ then $[B, B^\dagger] = 1$ to $O(n/N)$. So this scheme works but it does not make very efficient use of qubits. If we have K bosons to represent, we divide N into K parts so that $N_\alpha = N/K$ qubits represent 1 boson. We can find that similar commutation relations are obtained with similar errors.

Here's another approach from Ref. [2]. Consider a register of $N_p + 1$ qubits that can represent a maximum occupation of N_p bosons. We assign the following mapping:

$$|0\rangle \leftrightarrow |\uparrow\downarrow \ldots \downarrow\rangle \tag{1.3}$$
$$|1\rangle \leftrightarrow |\downarrow\uparrow \ldots \downarrow\rangle \tag{1.4}$$
$$\ldots \tag{1.5}$$
$$|N_p\rangle \leftrightarrow |\downarrow\downarrow \ldots \uparrow\rangle \tag{1.6}$$

The creation operator is

$$b_i^\dagger = \sum_{n=0}^{N_p} \sqrt{n+1}\, \sigma_-^{n,i} \sigma_+^{n+1,i} \tag{1.7}$$

However, these operators again do not exactly satisfy the bosonic commutation relations. (Note: the Macridin et al paper says that this scheme is also only suitable for fixed boson number, but from my read of Ref. [2] I don't see that it is true).

2 Alternate approach

2.1 Overview

The approach of Macridin et al improves on these prior schemes. In their scheme, the bosonic degrees of freedom of interest are represented as a finite set of harmonic oscillators. Through the mapping they describe, the low energy space of a harmonic oscillator is isomorphic to the low energy subspace of a finite size Hilbert space up to exponentially small error. The cost to include bosons scales as $O(Nn_x)$ where N is the number of harmonic oscillators and n_x is the number of qubits required for each harmonic oscillator. As we will see, in this scheme n_x can be as small as order 6-7, quite an improvement over the prior schemes that required a substantially greater number of qubits per harmonic oscillator so that the commutation relations were satisfied.

Let's assume we have a generic fermion-boson coupled Hamiltonian, the exact form of which is given in the paper. For now we consider a single harmonic oscillator (HO) with a Hamiltonian:

$$H_h = \frac{P^2}{2} + \frac{X^2}{2} \tag{2.1}$$

From our standard quantum mechanics course, the eigenfunctions of this Hamiltonian are Hermite-Gauss functions:

$$\langle x|\phi_n\rangle = \phi_n(x) = \frac{1}{\pi^{1/4}\sqrt{2^n n!}} e^{-x^2/2} H_n(x) \tag{2.2}$$

We can also define the function in the momentum basis $\langle p|\phi_n\rangle = \hat{\phi}_n(p)$. A useful fact is that HG functions are also eigenfunctions of the Fourier transform operator:

$$[F(\phi_n(x))](p) \equiv \hat{\phi}_n(p) = (-i)^n \phi_n(p) \tag{2.3}$$

By defining annihilation and creation operators b and $b^\dagger$ we can derive recurrence relations:

$$x\phi_n(x) = \frac{1}{\sqrt{2}}\left(\sqrt{n+1}\phi_{n+1}(x) + \sqrt{n}\phi_{n-1}(x)\right) \tag{2.4}$$

$$p\hat{\phi}_n(p) = \frac{i}{\sqrt{2}}\left(\sqrt{n+1}\hat{\phi}_{n+1}(p) - \sqrt{n}\hat{\phi}_{n-1}(p)\right) \tag{2.5}$$

2.2 Discretization

We now aim to discretize this representation of a bosonic state containing some number of quanta so that we can store it in a finite register of qubits. We make use of the fact that in both the coordinate and momentum representation, the HG functions decay to zero at large values of the argument. We can define a width $2L$ such that $|\phi_n(x)| \approx 0$ when $|x| > L$ for all $n < N_{ph}$ where N_{ph} is a cutoff number. Because $\phi_n(x)$ is an eigenfunction of the Fourier transform, we also have that $|\hat{\phi}(p)| \approx 0$ if $|p| > L$ and $n < N_{ph}$ (note that since the functions take non-dimensional arguments p and x can use the same cutoff).

Now we use the fact that functions with finite bandwidth (e.g. finite values of p in this context) can be represented as a discrete sum (Fourier series) with exponentially small error. In detail, the Whittaker-Shannon interpolation formula tells us that:

$$\phi_n(x) = \sum_{i=-N_x/2}^{N_x/2-1} \phi_n(x_i) u_i(x) \tag{2.6}$$

where N_x is some number that we need to figure out. The set of points x_i is $x_i = i\Delta$ (be careful that here i is an index, not the imaginary number). The spacing between the points x_i indicates the maximum frequency content in the signal. The functions $u_i(x) = sinc((x - x_i)/\Delta)$. Now let's get a relation between Δ and N_x. To restrict $x_i \in [-L, L]$ which we defined as the domain of the HG functions, we need $2L = N_x\Delta$. Therefore:

$$4L^2 = 2\pi N_x \rightarrow 2L = \sqrt{2\pi N_x} = 2\pi/\Delta \tag{2.7}$$

so that $\Delta = \sqrt{2\pi/N_x}$. We can derive from the Shannon-Nyquist sampling theorem that this expansion scheme also requires that $N_x > N_{ph}$ to avoid aliasing.

With this discretization, let's examine a finite subspace $\tilde{H}$ of the original HO Hilbert space spanned by the vectors $\{|x_i\rangle\}$. We can define kets on this space $|\chi_n\rangle$ such that $\langle x_i|\chi_n\rangle \equiv \sqrt{\Delta}\phi_n(x_i)$. The set $\{|\chi_n\rangle\}$ are orthonormal by the orthogonality of the HG functions. For momentum space, we also define:

$$\begin{aligned} \langle p_m|\chi_n\rangle &= \sqrt{2\pi\Delta}\hat{\phi}_n(p_m) && (2.8) \\ |p_m\rangle &= \frac{1}{\sqrt{N_x}} \sum_{i=-Nx/2}^{N_x/2-1} e^{ix_i p_m} |x_i\rangle && (2.9) \end{aligned}$$

where $p_m = m\Delta$. With the kets so defined on this discrete space, they obey the same recurrence relations as the original HG functions since the original relations were true for all arguments to the functions:

$$\begin{aligned} x_i \langle x_i|\chi_n\rangle &= \frac{1}{\sqrt{2}} \left(\sqrt{n+1} \langle x_i|\chi_{n+1}\rangle + \sqrt{n} \langle x_i|\chi_{n-1}\rangle\right) && (2.10) \\ p_m \langle p_m|\chi_n\rangle &= \frac{i}{\sqrt{2}} \left(\sqrt{n+1} \langle p_m|\chi_{n+1}\rangle + \sqrt{n} \langle p_m|\chi_{n-1}\rangle\right) && (2.11) \end{aligned}$$

We can now define discrete operators $\tilde{X}$ and $\tilde{P}$ just like the original X and P operators:

$$\begin{aligned} \tilde{X} |x_i\rangle &= x_i |x_i\rangle && (2.12) \\ \tilde{P} |p_m\rangle &= p_m |p_m\rangle && (2.13) \end{aligned}$$

Now, because the original X and P operators obeyed the recurrence relations and the canonical commutator $[X, P] = i$, so too must these discrete operators: $[\tilde{X}, \tilde{P}] = i$ in the space spanned by $|\chi_n\rangle$ and for $n < N_{ph}$ (recall the latter requirement follows from the cutoff of the extent of the HG functions).

The result of this analysis is that we now have the exact algebra of the original operators for the HO to within exponentially vanishing error (unlike the original schemes described at the beginning)!

2.3 Algorithm and qubit representation

We now figure out how to implement a digitized simulation of a fermionic-bosonic interacting Hamiltonian with the discrete space representing the bosonic representation described above. Trotterization is assumed to implement the propagator for the Hamiltonian. Each HO state is represented as a superposition of $N_x = 2^{n_x}$ states using n_x qubits:

$$|\phi\rangle = \sum_{j=0}^{2^{n_x}-1} \phi_j |x_j\rangle \tag{2.14}$$

Recall that $\tilde{X}|x_j\rangle = x_j|x_j\rangle$. Since x was originally stated to be in $[-L, L]$ but here j runs from $[0, 2^{n_x} - 1]$, the eigenvalue of the operator is $x_j = (j - N_x/2)\Delta$. A similar consideration arises for the $\tilde{P}$ operator; details are around p8 (note there is a typo in Eq 68, missing a $|p_n\rangle$ on the left-hand side).

Circuits for bosonic operators

Now let's consider the circuit. For this discussion, index n refers to a site label for a HO consisting of several qubits, not an individual basis state. I think the notation is confusing, so try to remember that $|x_n\rangle$ refers to the collection of basis states for an oscillator, $\{|x_j\rangle\}$ and $|x_j\rangle$ still refers to an individual computational basis state in a multiqubit register described by a sequence of binary values.

The first operator we need to implement is

$$e^{-i\theta\tilde{X}_n}|x_n\rangle = e^{-i\theta(x_n - N_x/2)}|x_n\rangle \tag{2.15}$$

Let's first consider a single basis state $|x_j\rangle$ and ignore the $N_x/2$ term (it is a classical phase we can track separately). With this assumption, the eigenvalue for $|x_j\rangle$ is $j\Delta$. We absorb Δ into the rotation angle θ, and so we need to get an eigenvalue for the exponential operator as $\exp(-ix_j\theta)$. Remember that for a multi-qubit register, the computational basis states can be written as $|0\rangle$, $|1\rangle$, $|2\rangle$, ... that correspond to sequences of binary numbers. For example, for a 2-qubit register, we have $|0\rangle \leftrightarrow |00\rangle$, $|1\rangle \leftrightarrow |01\rangle$, $|2\rangle \leftrightarrow |10\rangle$, and $|3\rangle \leftrightarrow |11\rangle$. Therefore the integer x_j can also be expressed in binary form. In the paper, x_j (interchangeable with integer j) is written in binary as:

$$x_j = \sum_{r=0}^{n_x-1} x_j^r 2^r \tag{2.16}$$

So to implement the exponential term on a single computational basis state $|x_j\rangle$, we would implement $\exp(-i\theta\sum_{r=0}^{n_x-1} x_j^r 2^r) = \exp(-i\sum_r \theta_r x_j^r)$ where $\theta_r \equiv \theta 2^r$.

Now let's consider the register of qubits denoted by $|x_n\rangle$ that contains coefficients times the computational basis states $|x_j\rangle$. Take as an example two qubits. If the right-most entry of the computational state in binary is 1 (e.g. in $|01\rangle$ and $|11\rangle$), we have $r = 0$ and so we need to apply $\exp(-i\theta_0)$. We just apply 1 to the other two basis states. The operator that implements this procedure to the Hilbert space of two qubits is:

$$\begin{pmatrix} 1 & & & \\ & e^{-i\theta_0} & & \\ & & 1 & \\ & & & e^{-i\theta_0} \end{pmatrix} = I \otimes T(\theta_0) \tag{2.17}$$

where $T(\theta_0)$ is a Z rotation matrix.

For $r = 1$, we need to apply $\exp(-i\theta_1)$ to states with the second-from-right entry as 1 ($|10\rangle$ and $|11\rangle$). That operator is:

$$\begin{pmatrix} 1 & & & \\ & 1 & & \\ & & e^{-i\theta_1} & \\ & & & e^{-i\theta_1} \end{pmatrix} = T(\theta_1) \otimes I \tag{2.18}$$

So the overall circuit diagram would have $T(\theta_0)$ on one qubit wire and $T(\theta_1)$ on another. If you extend this analysis to arbitrary number of qubits you hopefully can see you get the circuit given as Figure 5. Note that this circuit as written does not implement the $\exp(-i\theta N_x/2)$ term, hence the need to track this phase classically when implementing the term.

We can figure out the circuit for the next term in a similar way. The second term is $\exp(-i\theta \tilde{X}_n^2)$. This operator has an eigenvalue $(x_n - N_x/2)^2$. Following our earlier analysis, we write out x_n in binary and simplify:

$$\begin{aligned}
(x_n - 2^{n_x-1})^2 &= \left(\sum_{r=0}^{n_x-1} 2^r x_n^r - 2^{n_x-1}\right)^2 && (2.19)\\
&= 2^{2n_x-2} - \sum_{r=0}^{n_x-1} x_n^r (2^{2r} - 2 \times 2^{n_x+r-1}) + \sum_{r<s} x_n^r x_n^s 2^{r+s} \times 2 && (2.20)
\end{aligned}$$

where the first factor of 2 in the paranthesis is because there are two terms, and the second one is because the sum is restricted to $r < s$. Simplifying gives the expression in the caption of Figure 5.

The first term is a classical phase we can track separately, the second term we know how to do from the previous discussion, but the third term we have to think about. It says we should only implement a phase if both x_n^r and x_n^s are 1. This gate is just $diag(1, 1, 1, \exp(-i\theta_{r+s+1}))$ which we recognize as a controlled-Z rotation gate. Although the first set of gates can be done in parallel, this second set consists of $O(n_x^2/2)$ gates and some have to be done sequentially.

Now consider $\exp(-i\theta \tilde{X}_n \tilde{X}_m)$, representing the coupling of two distinct boson registers. We have:

$$(x_n - 2^{n_x-1})(x_m - 2^{n_x-1}) = \sum_{rs} x_n^r x_m^s 2^{r+s} - \sum_r (x_n^r + x_m^r) 2^{r+n_x-1} + 2^{2n_x-2} \tag{2.21}$$

The gates required for this circuit are similar to the previous case except the controlled gates must occur between qubits in different registers.

For the momentum operators, we need to perform a quantum Fourier transform to send $|x_n\rangle \to |p_n\rangle$, apply the operators as above, then perform an inverse QFT. Fermion operators are implemented as we have discussed in the rest of the class.

Circuits for fermion-boson operators

Fermion operators were covered in the previous lectures and are assumed to be handled by a Jordan-Wigner transform. Therefore, the last set of operators we need to implement are fermion-boson operators. These terms are of the form $(c_i^\dagger c_j + c_j^\dagger c_i)X_n$, $(c_i^\dagger c_j + c_j^\dagger c_i)P_n$, etc. Assume a JW transform was applied to the fermion operators. They then turn into:

$$c_i^\dagger c_j + c_j^\dagger c_i = \frac{1}{2}(X_i X_j + Y_i Y_n) Z_{i+1} ... Z_{j-1} \tag{2.22}$$

$$i(c_i^\dagger c_j - c_j^\dagger c_i) = \frac{1}{2}(Y_i X_j - X_i Y_n) Z_{i+1} ... Z_{j-1} \tag{2.23}$$

(More confusing notation above: X_i refers to the Pauli operator, not the boson operator X_n. I stick to the convention in the text to avoid further confusion).

Let's see how to implement the diagonal term corresponding to $\exp(-i\theta c_i^\dagger c_i)$. Remember how this term is implemented in a purely fermionic simulation - it becomes a phase shift gate $T(\theta)$ because $n_i = c_i^\dagger c_i = (1 - Z_i)/2$. If we now apply the fermion-boson operator to the combined fermion-boson register $|i\rangle |x_n\rangle$, we get:

$$e^{-i\theta c_i^\dagger c_i \tilde{X}_n} |i\rangle |x_n\rangle = e^{-i\theta c_i^\dagger c_i x_n} = T(\theta x_n) \tag{2.24}$$

So the gate to implement is still a phase shift gate, just with a different angle. Representing x_n in binary as before, we get $x_n \sum_r x_n^r 2^r$. We split the phase shift into a product of phase shifts controlled by x_n^r, leading to $n_x - 1$ controlled phase shift gates applied to the fermion register $|i\rangle$, controlled on the individual qubits of the boson register (see Fig 7 of the paper). Finally, we apply a phase shift to the fermion register using $T(-2^{n_x - 1}\theta)$ to get the second component of the eigenvalue $x_n - 2^{n_x-1}\theta$.

Now consider the off-diagonal hopping terms involving $c_i^\dagger c_j + c_j^\dagger c_i$. For the pure fermionic case, JW transform of the terms yields basis transform terms on qubits i and j (e.g. H for X operators), a string of CNOTS, a z rotation, a string of CNOTS, and the inverse basis transform. Similar logic shows that the same circuit is applied for the fermion-boson case except, as before, the $R_z(\theta)$ is replaced by $R_z(\theta x_n)$ exactly as we did before for the phase shift gate $T(\theta)$. As an example, here is a term:

$$e^{-i\theta(c_i^\dagger c_j + c_j^\dagger c_i)\tilde{X}_n} \approx \varsigma_i^\dagger \varsigma_j^\dagger e^{-i\theta Z_i..Z_j \tilde{X}_n} \varsigma_j \varsigma_i H_i H_j e^{-i\theta Z_i..Z_j \tilde{X}_n} H_j H_i \tag{2.25}$$

where $\varsigma_i = R_x(\pi/2)$ is a basis transform to the Pauli Y basis on fermion qubit i and the approximate comes from the Trotterization we used to approximate the sum of terms in the exponential. The circuit for this operator is very similar to the pure fermionic case excepting the change in the z rotation (see Fig 8 of the paper).

For the momentum boson-fermion operators, we have to do a QFT first, then this circuit, then inverse QFT.

2.4 Resource scaling

Under some mild assumptions described in the paper, the qubit number $n_x \sim O(\log(\ln \epsilon^{-1})$, meaning we need exponentially fewer qubits to represent a boson space with a given cutoff than for prior schemes. This favorable scaling arises because we are densely using all the states in superposition to represent the bosonic state. Another good scaling is that the circuit depth for implementing operations on a single HO scales polynomially as n_x^2.

A drawback to these nice results is that the circuit depth required for state preparation does scale exponentially in n_x: more precisely as $O(n_x^2 2^{n_x})$. That is because a generic quantum state like the dense one used to represent the bosons will always require exponential resources to prepare.

References

[1] L.-A. Wu and D. A. Lidar, "Qubits as parafermions", Journal of mathematical physics **43**, 4506–4525 (2002).

[2] R. D. Somma, G. Ortiz, E. H. Knill, and J. Gubernatis, "Quantum simulations of physics problems", in Quantum information and computation, Vol. 5105, edited by E. Donkor, A. R. Pirich, and H. E. Brandt, SPIE Proceedings (Aug. 5, 2003), p. 96.