Decomposing the J operator

April 2020

Rajiv Krishnakumar

We define the J operator as $\frac{1}{\sqrt{2}}\left(I^{\otimes N} + iX^{\otimes N}\right)$. If we take $N = 2$ we can see that J acts on the 4 basis states as described below:
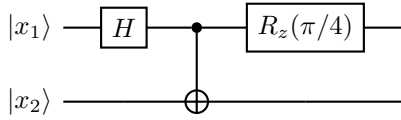
$$|00\rangle \rightarrow \frac{1}{\sqrt{2}}\left(|00\rangle + i\,|11\rangle\right) \tag{1}$$

$$|01\rangle \rightarrow \frac{1}{\sqrt{2}}\left(|01\rangle + i\,|10\rangle\right) \tag{2}$$

$$|10\rangle \rightarrow \frac{1}{\sqrt{2}}\left(|10\rangle + i\,|01\rangle\right) \tag{3}$$

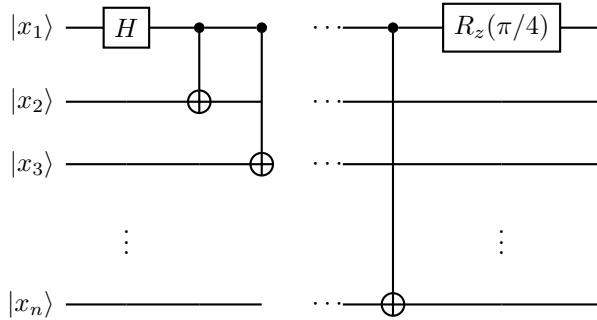$$|11\rangle \rightarrow \frac{1}{\sqrt{2}}\left(|11\rangle + i\,|00\rangle\right) \tag{4}$$

Creating a circuit to satisfy eqs. (1) and (2), we can use the following quantum circuit

In fact, for any n-qubit state that has its first qubit as $|0\rangle$ on which we want to act the J operator on, i.e.

$$|x_1 x_2 ... x_n\rangle \rightarrow \frac{1}{\sqrt{2}}\left(|x_1 x_2 ... x_n\rangle + i\,|\bar{x}_1 \bar{x}_2 ... \bar{x}_n\rangle\right) \tag{5}$$

where $|\bar{x}_i\rangle$ indicates *not* x_i, and where x_1 is set to 0, we can draw the circuit as


Now what about states that have their first qubit set to $|1\rangle$? To start with, let's look at trying to find a circuit for eqs. (3) and (4). What we can do is flip the first qubit, repeat the same circuit as we did for eqs. (1) and (2), and flip first the qubit back. This would look like
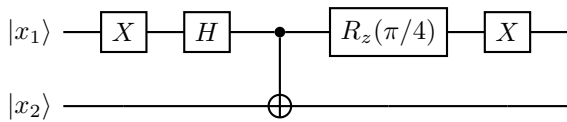

So now for any n-qubit state that has its first qubit as $|1\rangle$ on which we want to act the J operator on (which takes us back to eq. (5) except this time x_1 is set to 1), we can draw the circuit as
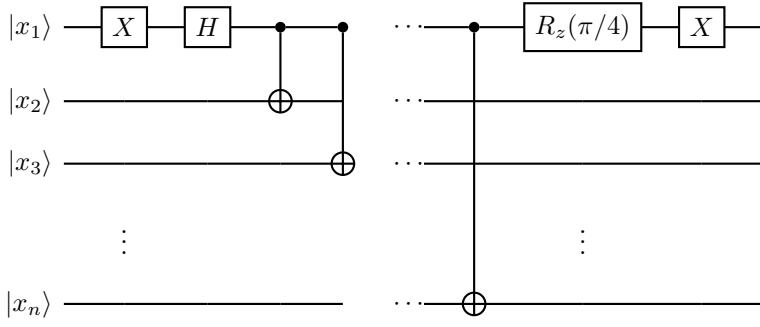
So now we want to make a single circuit that works for both cases: (1) when the first qubit is $|0\rangle$ *and* (2) when the first qubit is $|1\rangle$. One naive way to do it is to insert an ancilla qubit that stores the value of the first qubit with a CNOT gate. After that, you flip the ancilla qubit, and now you can implement the whole circuit for case (1) but with the additional feature that all the gates are controlled by the ancilla qubit (in addition to any other qubit they may be controlled by). This will lead to all the gates acting on the qubits if the first qubit was initially a $|0\rangle$, and do nothing if initially the first qubit was a $|1\rangle$. Then you flip the ancilla qubit back and finally implement the whole circuit for case (2) with once again the additional feature of all the gates being controlled by the ancilla qubit. This will lead to all the gates acting on the qubits if the first qubit was initially a $|1\rangle$, and do nothing if initially the first qubit was a $|0\rangle$. The full circuit looks like
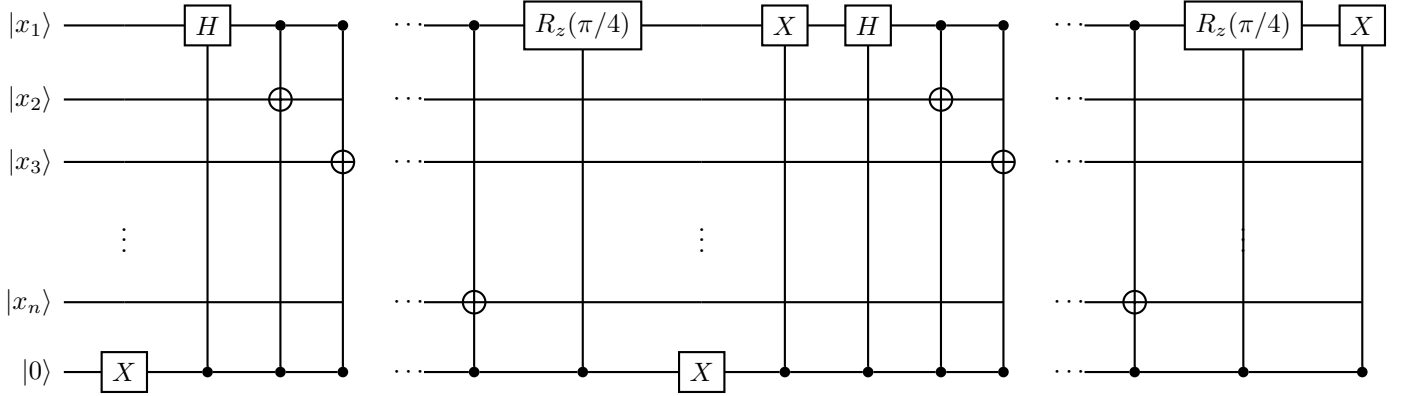
But can we be more efficient than this? Turns out we can a little bit. Firstly we observe that we can remove all control for the two X gates as long as all the gates in between are controlled by the unchanging ancilla qubit. This is because two X gates back-to-back equal the identity. And now we can make a few more subtle changes: we can remove the control on the first H gate, completely remove the middle H and $R_z(\pi/4)$ gates and move the final $R_z(\pi/4)$ gate to the very end of the circuit without a control. For now we can justify the last step of moving the $R_z(\pi/4)$ gate by saying that it cannot be between the two X gates since it is no longer controlled by the ancilla qubit. The final circuit now looks like
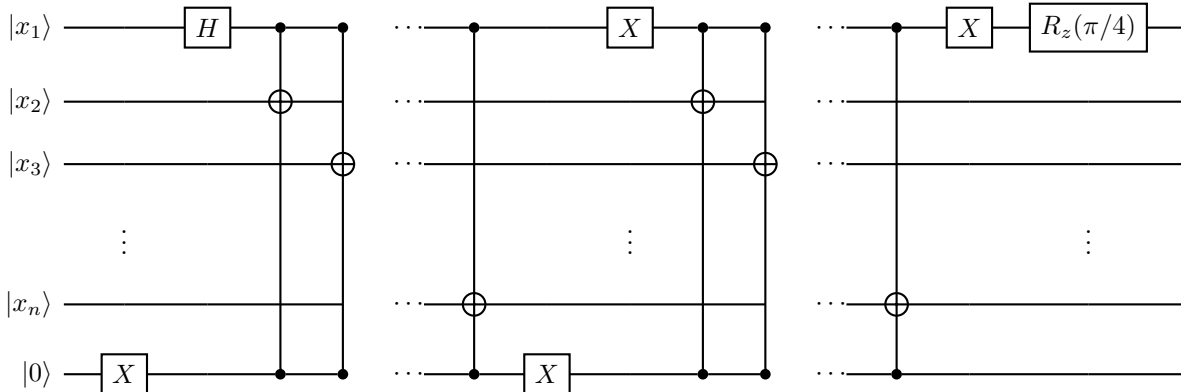
Does this slightly optmized circuit still do what we want it to do i.e. does it still fullfill eq. (5)? Let's go back to our two cases. We see that for the case where the first qubit is $|0\rangle$, everything in between the last two X gates can be ignored (and incidentally this means that the last two X gates acting on the first qubit can also be ignored too since they cancel each other out). This then brings us back to the original cicuit for case (1) above, which is exactly what we want. Similarly for case (2), we can ignore all the gates that are in between the *first* two X gates (the ones acting on the ancilla qubits). However this *almost* brings us back to the original circuit for case (2) above. The difference being the order of the H and first X gates are reversed, as well as the order of the final X and $R_z(\pi/4)$ gates acting on the first qubit are also reversed. It turns out that all this does is add a global phase. When worked out, this modifies eq. (5) to give

$$|x_1 x_2 ... x_n\rangle \rightarrow \frac{1}{\sqrt{2}} \left(-i\, |x_1 x_2 ... x_n\rangle + |\bar{x}_1 \bar{x}_2 ... \bar{x}_n\rangle\right)$$

$$\rightarrow -i\frac{1}{\sqrt{2}} \left(|x_1 x_2 ... x_n\rangle + i\, |\bar{x}_1 \bar{x}_2 ... \bar{x}_n\rangle\right)$$

for the case when the $x_1 = 1$. Because the J gate acts on all the gates in the system, adding this global phase does not change any of the measurement statistics. This is the decomposition that is currently implemented in the code.

This decomposition has the nice properties that the number of gates required scales linearly with the number of qubits and the number of ancillla qubits required is always 1 regardless of the total number of qubits. The downside is that it uses many Toffoli gates which (as all 3-qubit gates) are very noisy in real devices.