MSci Logbook

zcaplel

December 2016

1 Introduction

After implementing differential evolution on the average gate fidelity for a quantum half adder with the follow parameters, $NP = 16, F = 0.5, CF = 0.5, iters = 1881$, we end up with an average fidelity of -0.9999995094. This is incredibly close to one, making it a successful optimisation.

Taking the optimised parameters and constructing the hamiltonian, I wrote up a bit of code to test whether our quantum channel operator is producing the correct transformation to a state. This piece of code checks the fidelity between the gate operator and our quantum channel operation on the binary states.

```
for i in range(0, 2):
 for j in range(0, 2):
  for k in range(0, 2):

   psi0 = tensor(basis(2, i), basis(2, j), basis(2, k))
   psi = tensor(psi0, sin(0.847) * basis(2,1) + cos(0.847) * basis(2,0))

   rho = psi * psi.dag()
   trans_state = (U * rho * U.dag()).ptrace([0, 1, 2])
   trans_G_state = G * rho.ptrace([0, 1, 2]) * G.dag()
   print(fidelity(trans_G_state, trans_state))
```

Surprisingly, the output is not what I expected. The fidelity for each state is as follows:

```
000 ———> 0.999999638331591
001 ———> 0.9999998141131511
010 ———> 0.00024182248700191768
011 ———> 0.0002420332926820799
100 ———> 0.023159415820706835
101 ———> 0.012772408832805587
110 ———> 0.010165515430824378
111 ———> 0.022571237518917264
```

Since the fidelity for the transformation of each binary state isn't close to 1, the gate we have optimised is not the quantum half adder.

After looking more closely and evaluating the truth table for our quantum channel operator, it is clear that the operation being carried out is:

$$\mathcal{E} \approx \hat{I} \otimes c\hat{not}$$

This can be confirmed by running the above code with $G = \hat{I} \otimes c\hat{not}$, the results relatively close to 1.

```
000 ———> 0.999999638331591
001 ———> 0.9999998141131511
010 ———> 0.9999996850081154
011 ———> 0.9999997767620608
100 ———> 0.961607980196279
101 ———> 0.961851068093735
110 ———> 0.865061854836569
111 ———> 0.86491090049783
```

This is strange and perhaps it would help to break up the quantum half adder gate into its constituent gates and test what kind of quantum channel operation our optimisation algorithm produces for them.

The first constituent gate of the quantum half adder is the toffoli gate. After optimising our quantum channel operator to produce an average gate fidelity of -0.999992244829, the code above was used. Similarly to the quantum half adder, our implementation of the toffoli gate is not transforming the states in a way we would expect.

```
000 ———> 0.9999953034088173
001 ———> 0.999995192628406
010 ———> 0.9999961374040128
011 ———> 0.9999959413185942
100 ———> 0.9988618026910943
101 ———> 0.9691629684645193
110 ———> 0.0050802450287067025
111 ———> 0.005456970073779591
```

but instead, is simulating a gate operation similar to that of the identity operator:

$$\mathcal{E} \approx \hat{I}^{\otimes 3}$$

Running our test code for $G = \hat{I}^{\otimes 3}$ produces the following fidelities:

```
000 ———> 0.9999953034088173
001 ———> 0.999995192628406
010 ———> 0.9999961374040128
011 ———> 0.9999959413185942
100 ———> 0.9988618026910943
101 ———> 0.9691629684645193
110 ———> 0.9691319618109653
111 ———> 0.9989134992849659
```

Confirming that the channel that we optimised as a toffoli gate is acting very similarly to that of the identity over all states.

Next, I tried optimising the second constituent gate of the quantum half adder, the cnot gate acting on the first two qubits, this gate is represented like so,

$$G = c\hat{not} \otimes \hat{I}$$

Running our test code for the fidelity of the binary states of our optimised channel and the theoretical results of this gate,

```
000 ———> 0.9999924627823625
001 ———> 0.9999973385583868
010 ———> 0.00042670145991649153
011 ———> 0.0004975450199812413
100 ———> 0.006415713329803297
101 ———> 0.0071284549504984475
110 ———> 0.002153359874193084
111 ———> 0.012913201938587688
```

Again, this isn't correctly transforming the stated of $G = c\hat{not} \otimes \hat{I}$, instead we can see that it is implementing a different gate,

$$\mathcal{E} = \hat{I} \otimes c\hat{not}$$

Running our test code on $G = \hat{I} \otimes c\hat{not}$ confirms this.

```
000 ———> 0.9999924627823625
001 ———> 0.9999973385583868
010 ———> 0.9999982471268771
011 ———> 0.9999996168954642
100 ———> 0.9625763519513799
101 ———> 0.9625233303312937
110 ———> 0.7514804881305945
111 ———> 0.7515064173903468
```

Although these values aren't what we want, at least the error is consistent, perhaps allowing us to identify what went wrong in the code. A quantum half adder can be written in the following way,

$$G_{QHA} = (c\hat{not} \otimes \hat{I})\hat{T}$$

Plugging in our incorrect implementations for $c\hat{not} \otimes \hat{I}$ and $\hat{T}$, we can see that the incorrect theoretical value agrees with our first implementation.

$$G_{QHA} = (\hat{I} \otimes c\hat{not})\hat{I}^{\otimes 3} = \hat{I} \otimes c\hat{not}$$

This allows us to deduce that the calculation with the constituent gates for the quantum half adder is correct, and there is something fundamentally wrong with the fidelity calculation within the code itself.