Quantum Computing PHYS-541, Project 7

Teacher : vincenzo.savona@epfl.ch
Assistant : sara.alvesdossantos@epfl.ch, david.linteau@epfl.ch, shao.chiew@epfl.ch

Shor's factoring algorithm

We have studied in class *Shor's factoring algorithm.*

The goal of the project is:

1. Read and understand the chapter on Shor's factoring algorithm in Nielsen & Chuang's book, and possibly on other sources, and present to a sufficient level of detail how Shor's algorithms works. Explain in particular how the algorithm relies on the order or period finding problem.

2. Devote part of your presentation to the modular exponentiation, which defines the task to be carried out by the oracle. What may be a systematic way to write a modular exponentiation circuit starting from a classical boolean circuit and using reversible computing? Starting from this result, how much room there is for improvement? There are basically two ways of writing modular multiplication or exponentiation circuits: (i) bottom-up [1, 2, 3, 4] and (ii) compiled [5, 6]. Compiled means that you already know the results of the arithmetic operations, and you just code a circuit that transforms each input into the known output. Bottom-up is the way Shor's algorithm should be written for factoring numbers that we can't factor otherwise. It consists in coding the modular multiplication from a code for modular sum, and the modular exponentiation from the modular multiplication. For this project, you will **code and compare the two approaches** in terms of circuit depth and number of required qubits.

3. Implement Shor's algorithm on the IBM-Q Qiskit platform (on the QASM simulator). The specific task is to implement the algorithm for factoring $N = 15$, as seen in class, and then an algorithm for factoring $N = 21$.

4. Study the performance of the algorithm in presence of (simulated) noise. In particular, what is the statistical error of the $N = 21$ algorithm, compared with the one for $N = 15$ algorithm? Discuss the feasibility of larger factoring tasks on current quantum hardware.

References

[1] Stephane Beauregard. Circuit for Shor's algorithm using 2n+3 qubits, February 2003. arXiv:quant-ph/0205095.

[2] Steven A. Cuccaro, Thomas G. Draper, Samuel A. Kutin, and David Petrie Moulton. A new quantum ripple-carry addition circuit, October 2004. arXiv:quant-ph/0410184.

[3] Thomas G. Draper. Addition on a Quantum Computer, August 2000. arXiv:quant-ph/0008033.

[4] Igor L. Markov and Mehdi Saeedi. Constant-Optimized Quantum Circuits for Modular Multiplication and Exponentiation, April 2015. arXiv:1202.6614 [cs].

[5] Omar Gamel and Daniel F. V. James. Simplified Factoring Algorithms for Validating Small-Scale Quantum Information Processing Technologies, November 2013. arXiv:1310.6446 [quant-ph].

[6] Robert L. Singleton Jr. Shor's Factoring Algorithm and Modular Exponentiation Operators. Quanta, 12:41–130, September 2023. arXiv:2306.09122 [quant-ph].