

PHYS-541 Exam: Questions

Prof. Vincenzo Savona

December 24, 2024

Chapter 1

Questions for the projects

1.1 Project 01

- Explain the cross-entropy benchmark (XEB) assuming I don't know anything about it, but I understand quantum computing very well.
- Can you execute the cross-entropy benchmark on a circuit other than a random quantum circuit? What are the assumptions underlying the XEB?
- Does it make a difference if you use C-NOT or F-SIM two-qubit gates in the random quantum circuit? Why?

1.2 Project 02

- The time evolution of the density operator in this problem is not unitary. How do you implement it on a quantum computer, which can only execute unitary operations?
- The quantum phase estimation is used to estimate an eigenvalue. Here the eigenvalue of the NESS is zero, so it is already known. What do you use QPE for?
- How do you make sure that the input state to the QPE algorithm has a significant overlap with the NESS state?
- How did (simulated) noise affect even the smallest instance of the algorithm?

1.3 Project 03

- For the 1st order Suzuki-Trotter (ST) scheme, how does the ST error depend on the total time t and on the number of ST steps n ?
- How does the number of gates in the 1st order ST scheme depend on the total time t and on the desired ST accuracy ϵ ?
- Give the expressions for 1st and 2nd-order ST schemes.
- How does the ST error depend on t and n for the 2nd-order ST scheme?

- Can you extrapolate, intuitively, from the 1st and 2nd order ST schemes, how the ST error will depend on t and n in the limit of infinite order ST scheme?
- Same question as above, for the total number of gates.
- Discuss the relative importance of ST error and noise error. Do they scale the same as a function of total time t and number of qubits n ?

1.4 Project 04

- Suppose you can only use circuit folding. Assume an error rate of 0.1% per gate (on average). At what circuit size will error mitigation become ineffective? Measure circuit size in number of qubits n times depth d .
- With the same assumptions as above, can layer folding improve the efficiency of error mitigation?
- Look at your result as a function of the number of foldings. Is it realistic? How much noise error would you expect (e.g. from the arguments of the two previous questions)?
- Which among the possible zero-noise extrapolation functions should be the most correct? Do you have any physical intuition on this?
- Do you know any other error mitigation technique? Can you think of any?

1.5 Project 05

- Assume a quantum circuit with n qubits and depth d . In general, assuming $v = n \times d$ constant, is it easier to simulate circuits with large n and small d , or small n and large d , or other? How does this depend on the simulation method?
- Discuss how the computational complexity of the simulation must depend on n and d assuming that you use the most effective approach to simulate the circuit. Use physical arguments.
- You know that a Clifford circuit can be simulated efficiently in polynomial time. How would you simulate a large circuit which is made only of Clifford gates, except for one T-gate in the middle of the circuit. How can you simulate it efficiently? Are the methods you studied for this project also efficient?
- Same question but for t T-gates instead of one. How does the complexity scale with t ?
- Explain how you can sample bitstrings from the output of your simulation.
- Can you imagine a quantum circuit where simulating the final state is computationally hard but sampling from the corresponding measurement probabilities is computationally treatable?

1.6 Project 06

- How does the Grover search algorithm achieve a quadratic speedup compared to classical search methods, and what is the significance of the Grover angle in this context?
- Explain the relationship between Grover's search algorithm and quantum computational complexity, particularly in the context of NP problems. How does it highlight the limitations of quantum speedup?
- Describe how you implemented the oracle for Grover's algorithm in Qiskit. What challenges did you face when creating the oracle for $m = 1$ or $m = 2$, and how did you overcome them?
- What are the advantages and disadvantages of using ancilla qubits in constructing the oracle? Could you successfully implement an oracle without using ancilla qubits, and if so, how?
- How does noise affect the performance of Grover's algorithm as the size of the database (n) increases? Discuss the role of the depth of the Grover oracle in this analysis.

1.7 Project 07

- How does Shor's algorithm solve the factoring problem, and what is the link between factoring and the order or period finding problem?
- Explain the role of modular exponentiation in Shor's algorithm. How does this component contribute to the quantum advantage in factoring large numbers? What would be the algorithm complexity if you used modular multiplication instead?
- How did you construct the modular exponentiation circuit? Discuss any optimizations you considered or implemented and their impact on the algorithm's efficiency.
- Modular exponentiation can be efficiently computed classically. QFT, for separable input state, has separable output state. Why can't we simulate Shor's algorithm efficiently on a classical computer?
- Compare the statistical errors observed in the implementation of Shor's algorithm for $N = 15$ and $N = 21$. How does error affect the result in both cases?
- Based on your analysis, discuss the feasibility of using current quantum hardware for factoring larger numbers. What are the main challenges?

1.8 Project 08

- Explain the equivalence between the digital error model and the noisy quantum channel description.
- What are the Knill-Laflamme conditions for correctable errors, and how do they ensure the reliability of quantum error correction codes?

- How did you derive the way to encode the X and H gates on Shor's code?
- Describe the circuits you implemented for detecting and correcting X and Z errors, and explain how these circuits handle linear combinations of Pauli errors.
- What is the difference between a non-degenerate and a degenerate error correction code?
- Can you extend Shor's code to a code with larger distance d ?
- When introducing simulated noise in the QASM simulator, can you reach break-even, i.e. logical error rate lower than the physical error rate?

1.9 Project 09

- What is the stabilizer formalism, and how does it provide a systematic way to define quantum error correction codes (QECCs)?
- Explain the basic principles of fault-tolerant quantum computing. How does the concept of a threshold theorem contribute to the feasibility of fault tolerance?
- Describe your implementation of the three-qubit repetition code in Qiskit. How did you construct the encoding and decoding circuits, and how did you verify their correctness?
- From the comparison between the three-qubit and the nine-qubit codes, how do you expect the total number of gates to scale with the degree of concatenation?
- What is a stabilizer state? What is the difference between a stabilizer state and a stabilizer code?
- When simulating X -errors on physical qubits, how did you demonstrate the ability of the concatenated code to correct up to two errors in the same group of three qubits? Provide examples from your implementation.
- Does your code work also if you use the QASM noisy simulator, instead of a simple random X gate? Why?

1.10 Project 10

- What are variational quantum algorithms (VQAs), and why are they particularly suited for NISQ (Noisy Intermediate-Scale Quantum) hardware?
- How many one-shot measurements are required to reach a given accuracy ϵ in estimating the expectation value of an observable on the variational state?
- Based on the previous answer, how do you expect a variational algorithm to scale with the size of the problem? Is it exponential or polynomial scaling?
- Justify your choice of the variational ansatz.
- From physical arguments, how many layers should a variational ansatz have to describe accurately a quantum state of n qubits? Why?

- What is a barren plateau? When does it occur more often?
- How did you evaluate the robustness of your chosen VQA against noise? Describe the specific noise model you simulated and its impact on the algorithm's performance.

1.11 Project 11

- Discuss the effect of (simulated) noise on both the variational quantum dynamics and the Suzuki-Trotter digital quantum dynamics.

1.12 Project 12

- The questions listed in the text of the project are sufficient.

Chapter 2

Questions for the course

2.1 Basics of Quantum Mechanics

- Write the time-dependent Schrödinger equation for a given Hamiltonian $\hat{H}$.
- Write the unitary time-evolution operator $\hat{U}(t)$ for a given Hamiltonian $\hat{H}$.
- Assume a quantum bit in an arbitrary state $|\psi\rangle$. How do you write the probability of measuring the values $x = \pm 1$ when measuring the observable X ? (i.e. the Pauli matrix X)
- Assume you have a quantum computer made of *qutrits*, where a qutrit is a quantum system in a Hilbert space of dimension $d = 3$. What is the dimension of the Hilbert space of n qutrits?
- Define the Bloch sphere and explain how it describes the state of a quantum bit.

2.2 Digital Quantum Computation

- How do you write a C-Z gate using H and C-NOT gates?
- How do you write a C-U gate, where U is an arbitrary 1-qubit unitary, using only 1-qubit unitaries and C-NOTs?
- How do you write a doubly controlled $C^2 - U$ gate, where U is an arbitrary 1-qubit unitary, using only controlled 1-qubit unitaries and C-NOTs?
- How do you write a multiply controlled 1-qubit unitary $C^n - U$ using more elementary operations?
- Is it computationally more complex to express an arbitrary n -qubit unitary U in terms of 1-qubit unitaries and C-NOTs, or to express an arbitrary 1-qubit unitary U in terms of a universal set of 1-qubit gates $\{H, S, T\}$?
- How do you carry out a measurement of an observable $\hat{O}$ on the output state of a quantum circuit? Make an example with the observable X .
- Explain the principle of implicit measurement.

- Explain the principle of deferred measurement.

2.3 Quantum algorithms

- Explain the phase-kickback mechanism. Give an example, e.g. in Deutsch's algorithm.
- How would you write the oracle circuit of Deutsch's algorithm, when given a specific function $f(x)$?
- Illustrate Deutsch-Jozsa algorithm.

2.4 Computational complexity

- Provide examples of computational tasks in P , NP , and NP -Hard.
- Can you give examples of computational problems that are in BPP but not in P ?
- Can you give examples of computational problems that are in BQP but not in BPP?
- Define the BQP and QMA complexity classes and give examples of computational problems belonging to these classes.
- Explain the importance of Deutsch-Jozsa, Bernstein-Vazirani, and Simon algorithms in the context of quantum computational complexity.

2.5 QFT and Quantum Phase Estimation

- Define the Quantum Fourier Transform. Give both the expression as a linear superposition, and the one as a product state.
- If the input is a computational basis state, is the output of the QFT entangled?
- Do you need an entangled input, to obtain an entangled output state from QFT?
- Can you give me an approximate gate count for the QFT?
- How would you simplify the QFT algorithm to make it approximate but using less gates?
- Explain the goal of the Quantum Phase Estimation (QPE) algorithm and why it brings a computational advantage.
- How do you choose the number of qubits t of the register where the phase estimate is output?
- What happens in the QPE if the number of qubits t in the first register is smaller than the total number of binary digits needed to express the phase?
- What happens in the QPE if the input state is not exactly an eigenstate of the unitary operator U ?

2.6 Shor's factoring algorithm

- How does Shor's algorithm solve the factoring problem, and what is the link between factoring and the order or period finding problem?
- Explain the role of modular exponentiation in Shor's algorithm. How does this component contribute to the quantum advantage in factoring large numbers? What would be the algorithm complexity if you used modular multiplication instead?
- Modular exponentiation can be efficiently computed classically. QFT, for separable input state, has separable output state. Why can't we simulate Shor's algorithm efficiently on a classical computer?
- How does Shor's algorithm use QPE? Is the input state an eigenstate of the modular product?
- What is the probability of obtaining a valid factor from a single execution of Shor's algorithm?
- Give a step-by-step example of how Shor's algorithm would factor 15.
- Discuss the feasibility of using current quantum hardware for factoring larger numbers. What are the main challenges?

2.7 Grover's algorithm

- How does the Grover search algorithm achieve a quadratic speedup compared to classical search methods, and what is the significance of the Grover angle in this context?
- Explain the relationship between Grover's search algorithm and quantum computational complexity, particularly in the context of NP problems. How does it highlight the limitations of quantum speedup?
- How do you apply Grover's algorithm in the case where the database contains more than one solution to the search problem?
- How would you build an oracle for Grover's algorithm? What are the advantages and disadvantages of using ancilla qubits in constructing the oracle? Could you successfully implement an oracle without using ancilla qubits, and if so, how?

2.8 Digital Quantum Simulation

- Explain the Suzuki-Trotter decomposition and how it applies to the digital simulation algorithm.
- Explain the difference between 1st and 2nd order Suzuki-Trotter decomposition.
- How does the time-step error scale with the order of the ST decomposition?
- How does the number of gates scale with the order of the ST decomposition?

- It the computational complexity of the digital quantum simulation polynomial or exponential in the number of qubits? Under which conditions?

2.9 Density operator formalism

- What are the defining properties of the density operator?
- How to compute the density operator of a subsystem of a larger system?
- Compute the partial trace in the case of simple quantum states (e.g. two-qubit entangled states)
- Write the spectral representation of a density operator.
- What is the difference between a pure state and a statistical mixture?
- How does a density operator evolve in time in the case of an isolated system?
- Define a quantum channel and its properties. Define the Kraus operators and their properties.
- Define and describe the depolarization channel.
- Same for the phase-damping channel and the amplitude-damping channel.

2.10 Quantum error correction

- Explain the equivalence between the digital error model and the noisy quantum channel description.
- What are the Knill-Laflamme conditions for correctable errors, and how do they ensure the reliability of quantum error correction codes (QECCs)?
- Describe the simple repetition code and Shor's quantum error correction code.
- What is the difference between a non-degenerate and a degenerate error correction code?
- Define the distance d of a QECC.
- What is a stabilizer code? How can you use the stabilizer formalism to define a QECC?
- How to distinguish, within the stabilizer formalism, the error syndromes, the correctable errors, and the uncorrectable errors?

2.11 Fault-tolerant quantum computing

- Define fault-tolerant quantum computing.
- Define the Clifford group. What is its importance for fault tolerance?
- State the Gottesman-Knill theorem.
- Explain the notions of error propagation and transversal gates.

- Define the threshold of a QECC.
- What is the fundamental result emerging from the threshold theorems and why is it important for quantum computing?

2.12 The variational quantum eigensolver

- Explain the basic principle of VQE.
- What are the two main broad classes of variational Ansätze? What are their fundamental differences?
- How does the number of shots (i.e. circuit executions and measurements) needed to achieve accuracy ϵ scale with ϵ ? How does it scale with the number of Pauli terms in the Hamiltonian?
- Give an idea of what is a barren plateau.

2.13 The quantum approximate optimization algorithm

- Describe the idea of quantum annealing.
- What do *adiabatic* and *diabatic* mean in the context of quantum annealing?
- Explain the basic principle of QAOA.
- Explain the relation between analog quantum annealing and the QAOA algorithm.