



Dipartimento di Fisica Giuseppe Occhialini

Tesi di Laurea Triennale in Fisica

Esperimenti su singolo qubit:

implementazione e caratterizzazione di un quantum classifier

Candidato: Galizzi Federico

Matricola: 839936

E-mail: f.galizzi1@campus.unimib.it

Recapito telefonico: 331 3170467

Relatore: Dr. Giachero Andrea

Correlatore: Dr. Carrazza Stefano

Anno Accademico 2020-2021, sessione del 22-23-24 settembre

Riassunto:

Nel 1981, alla conferenza “*Physics of Computation Conference*”, Richard Feynman affermò la necessità di utilizzare computer basati sui principi della meccanica quantistica per poter indagare al meglio i fenomeni tipici del mondo atomico e subatomico. Computer quindi in grado di sfruttare effetti quali *entanglement*, interferenza e sovrapposizione di stati che l’informatica classica non utilizzava. Nacquero così i concetti di computer quantistico e *quantum computing*. Da allora, fisici, matematici, informatici e ingegneri hanno avuto a che fare con notevoli sfide da affrontare, tra le quali: le difficoltà di costruzione di sistemi così sofisticati e la stesura di algoritmi che possono effettivamente essere svolti da *device* quantistici. A quarant’anni da quella storica conferenza, IBM, che fu tra le promotrici dell’evento, offre un servizio cloud gratuito per imparare ad utilizzare computer quantistici, ad interagire con questi e per testare algoritmi che si basano sui principi del quantum computing. Il lavoro di tesi qui presentato è stato possibile grazie a questo servizio.

Il principale argomento di studio di questa tesi è il bit quantistico, comunemente chiamato *qubit*. Nel primo capitolo viene introdotto il concetto di qubit, presentandolo inizialmente come oggetto puramente matematico; astrazione con la quale i pionieri di questa nuova branca dell’informatica hanno lavorato. Tra essi si ricordano Peter Shor, David Deutsch e Richard Josza a cui si devono gli omonimi algoritmi verificati anni dopo la loro ideazione. Già su piano teorico, infatti, è possibile discutere circa cosa sia possibile e cosa impossibile fare con un qubit. In particolare, si presterà attenzione a come trarre vantaggio da un sistema che può essere in una qualsiasi combinazione di stati ma dal quale si può estrarre informazione classica solo attraverso un’operazione di misura; operazione che è assunta essere proiettiva da uno degli assiomi della meccanica quantistica. Proprio come per i bit tradizionali, gli unici risultati ottenibili da una singola misura applicata ad un qubit sono 0 e 1; indicati anche con $|0\rangle$ e $|1\rangle$, in notazione di Dirac, per distinguerli dal caso classico.

A questa trattazione seguirà un’altra che più si avvicina alla realizzazione di un computer quantistico elencando e commentando i criteri di Davide DiVincenzo che, nel 2000, propose i requisiti minimi che un computer dovesse avere per essere adatto al quantum computing. Questi criteri, da allora, non sono stati alterati e negli anni successivi sono stati soddisfatti da diverse aziende (Google, IBM, Rigetti, D-Wave) con dispositivi che si basano su tecnologie come: difetti nei reticoli di diamanti, trappole di ioni, risonanza magnetica nucleare in molecole (*NMR*), quantum dots, fotoni e circuiti superconduttori. Di questi ultimi verrà fornita l’intera trattazione dell’Hamiltoniana del relativo sistema. Si è deciso di porre particolare attenzione ai qubit superconduttori in quanto su questa tecnologia si basano i computer di IBM, dunque anche il *backend* utilizzato qui per gli esperimenti. La quale scommette sui qubit superconduttori in quanto sono i più promettenti in materia di scalabilità, ossia sembrano essere i sistemi che permetteranno la costruzione di computer con maggior numero di qubit. Recentemente IBM ha annunciato che rilascerà nel 2023 un backend -*ibmq-condor*- costituito di 1121 qubit. Attualmente, il loro computer di punta ne ha 65.

A termine del capitolo 1 si introduce la *IBM Quantum Experience*, il servizio cloud già citato, e i due *framework* Qiskit e QiskitPulse che contengono le librerie necessarie per la scrittura e l’esecuzione di algoritmi su computer quantistici reali. Più precisamente, Qiskit serve per la costruzione di circuiti quantistici atti a svolgere i programmi tipici del quantum computing. QiskitPulse, invece, permette la creazione di programmi a livello di impulsi e consente quindi di avere un maggiore controllo sui qubit. Come usare questa ulteriore possibilità viene mostrato nel capitolo 2, con l’esperimento di caratterizzazione di un qubit; il quale prevede l’utilizzo di impulsi nella banda delle microonde per trovarne la frequenza di risonanza attraverso tecniche di spettroscopia. Trovata questa, si procede con l’ottimizzazione dei parametri d’impulso necessari per implementare un *X gate* (l’analogo della porta NOT dell’informatica classica), grazie al quale si può far passare il qubit dallo stato fondamentale a

quello eccitato e viceversa. Successivamente si mostrano gli esperimenti necessari per la stima di T_1 e T_2 (i tempi caratteristici di rilassamento e decoerenza), due parametri fondamentali per soddisfare i criteri di DiVincenzo.

Il capitolo 3 si configura come lavoro di ottimizzazione dell'utilizzo di un qubit. Si studia quindi come poter far transire un qubit da uno stato ad un altro in maniera efficiente; un lavoro necessario in termini di affidabilità degli algoritmi di quantum computing.

Nel quarto capitolo si abbandona l'ambito dell'interazione con l'hardware a livello di impulsi per proporre un esperimento di implementazione di quello che potrebbe essere definito “*quantum classifier*”. Sostanzialmente, viene descritto come sia possibile costruire un programma che, attraverso tecniche di *machine learning*, è in grado di dare una stima dello stato in cui si trova un qubit dopo un'arbitraria serie di rotazioni applicate ad esso basandosi esclusivamente sui risultati di circuiti quantistici. Compito non banale, in quanto i risultati dei circuiti quantistici restituiscono le ampiezze di probabilità di misurare $|0\rangle$ oppure $|1\rangle$. L'esperimento è stato proposto con l'intento di mostrare i possibili vantaggi che il quantum computing può dare anche solo con un qubit, sebbene diversi libri di testo e articoli divulgativi individuano nell'entanglement e nell'interferenza (fenomeni che richiedono un numero plurale di qubit) i principali vantaggi dell'informazione quantistica nei confronti di quella classica.



Giuseppe Occhialini Physics Department

Physics Bachelor Thesis

Single-qubit experiment:

Implementation and characterization of a quantum classifier

Candidate: Galizzi Federico

Identification number: 839936

E-mail: f.galizzi1@campus.unimib.it

Phone number: 331 3170467

Advisor: Dr. Giachero Andrea

Co-Advisor: Dr. Carrazza Stefano

Year 2020-2021, 22-23-24 September session

Abstract:

It was 1981, at the *Physics of Computation Conference*, when Richard Feynman talked about the necessity to create computers based on the principles of the quantum mechanics to better investigate atomic and subatomic phenomena. Computers able to take advantage of effects as entanglement, interference and superposition of states; effects unknown to the Informatics of the time. In this way the idea of quantum computer and quantum computing took place. Since then, physicists, mathematicians, engineers and computer scientists had to deal with harsh tasks; for example: the difficulties of building extremely sophisticated systems and to conceive algorithms which could be executed on real quantum device. Forty years later, one of the main promoters of that historical conference, IBM, offers a free cloud service to learn to use quantum computers, to interact with them and to test algorithms based on quantum computing concepts. The work presented here was possible thanks to this service.

The main topic of this thesis is the quantum bit, also known as *qubit*. In the first chapter, the concept of qubit is introduced as a mere mathematical object; such an abstraction was used by the pioneer of this new branch of Science. In particular, I would like to mention Peter Shor, David Deutsch and Richard Jozsa whose algorithms are some of the most famous ones. In fact, just from a theoretical point of view, there is the chance to argue about what is possible and what is impossible to do with a qubit. How to take advantage from a system which could be in any one of an infinite number of linear combinations of states is a topic of particular interest. It is convenient to remember here, that a “measure” in quantum mechanics is a projective operation; this is possibly the most bizarre axiom of this subject. In the same fashion of classical bits, we can obtain just two different results from applying a measurement to a qubit: 0 or 1, which are commonly indicated with $|0\rangle$ and $|1\rangle$ in the Dirac’s notation. After such a theoretical approach, a more practical topic is discussed: the DiVincenzo criteria; all of whom establish the features needed for a device to be considered a quantum computer. Since 2000, these criteria have never been changed and some companies, such as Google, IBM, Rigetti and D-wave, have been able to fulfill all the criteria’s requests. They used quantum dots, NV diamonds, NMR in molecules, ion traps, photons and superconducting circuits. About the latter of this list will be provided a complete explanation of the Hamiltonian which describe the system. The choice of this deep interest for the superconducting qubit has been suggested by the fact that they are the ones used by IBM and so is the backend employed in the experiments of this work. Superconducting quantum computers seem to be the most scalable system; indeed, IBM has recently announced that could be able to build a 1121 qubits computer in 2023; its name should be *ibmq-condor*. At the moment, their computer with the largest number of qubits is made of 65.

At the end of chapter 1, IBM Quantum Experience, Qiskit and QiskitPulse are introduced. These two frameworks provide the necessary Python libraries to write and execute algorithms on real quantum devices. In particular, Qiskit is useful for the construction of the quantum circuits on which algorithms can be tested; QiskitPulse, instead, allows the creation of pulse-level schedules; thanks to this approach, the user is enabled to have a greater control on the qubits. In chapter 2 it is showed how to use these features; it will be illustrated a characterization experiment: finding the resonance frequency of the circuit employing spectroscopy methods, and measuring characteristic times of relaxation and decoherence (T_1 and T_2).

In chapter 3 an optimization process will be described. It is here discussed how to speed up the operation applied to a qubit (quantum gates) and how to create high-fidelity pulse to implement such gates. This kind of task is essential to be able to run algorithms on a quantum computer.

In chapter 4 it is provided an example of quantum classifier. Roughly speaking, a program based on machine learning techniques, which is able to estimate the state of a qubit after an arbitrary sequence

on gate applications. It can be an example of the advantages offered by quantum computers, which are even more convenient when entanglement and interference are properly used.