



ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ  
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ  
ΔΠΜΣ ΕΠΙΣΤΗΜΗ ΔΕΔΟΜΕΝΩΝ ΚΑΙ ΜΗΧΑΝΙΚΗ ΜΑΘΗΣΗ

Εισαγωγή στους Κβαντικούς Υπολογιστές,  
στους Κβαντικούς Αλγορίθμους για Μηχανική  
Μάθηση, σχεδιασμός και υλοποίηση ενός  
Κβαντικού kmeans

ΜΕΤΑΠΤΥΧΙΑΚΗ ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

ΤΟΥ

Κωνσταντίνου Μπιτσάκου

Διπλωματούχου Ηλεκτρολόγου Μηχανικού &  
Μηχανικού Υπολογιστών Ε.Μ.Π.

Επιβλέπων: Νεκτάριος Κοζύρης, Καθηγητής Ε.Μ.Π.

Αθήνα, Οκτώβριος 2021





Εθνικό Μετσόβιο Πολυτεχνείο  
Σχολή Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών  
ΔΠΜΣ Επιστήμη Δεδομένων και Μηχανική Μάθηση

# Εισαγωγή στους Κβαντικούς Υπολογιστές, στους Κβαντικούς Αλγορίθμους για Μηχανική Μάθηση, σχεδιασμός και υλοποίηση ενός Κβαντικού kmeans

## ΜΕΤΑΠΤΥΧΙΑΚΗ ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

του

**Κωνσταντίνου Μπιτσάκου**

Διπλωματούχου Ηλεκτρολόγου Μηχανικού και Μηχανικού Υπολογιστών Ε.Μ.Π.

**Συμβουλευτική Επιτροπή:** Νεκτάριος Κοζύρης, Καθηγητής  
Γεώργιος Γκούμας, Καθηγητής  
Διονύσης Πνευματικάτος, Καθηγητής

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την Ιούνιος 2022.

(Υπογραφή)

(Υπογραφή)

(Υπογραφή)

.....  
Νεκτάριος Κοζύρης  
Καθηγητής  
Ε.Μ.Π.

.....  
Γεώργιος Γκούμας  
Καθηγητής  
Ε.Μ.Π.

.....  
Διονύσης Πνευματικάτος  
Καθηγητής  
Ε.Μ.Π.

Αθήνα, Δεκέμβριος 2021

(Υπογραφή)

.....  
**Κωνσταντίνος Μπιτσάκος**

Διπλωματούχος Ηλεκτρολόγος Μηχανικός και Μηχανικός Υπολογιστών Ε.Μ.Π.

Copyright ©–All rights reserved Κωνσταντίνος Μπιτσάκος, 2021.

Με επιφύλαξη παντός δικαιώματος.

Απαγορεύεται η αντιγραφή, αποθήκευση και διανομή της παρούσας εργασίας, εξ' ολοκλήρου ή τμήματος αυτής, για εμπορικό σκοπό. Επιτρέπεται η ανατύπωση, αποθήκευση και διανομή για σκοπό μη κερδοσκοπικό, εκπαιδευτικής ή ερευνητικής φύσης, υπό την προϋπόθεση να αναφέρεται η πηγή προέλευσης και να διατηρείται το παρόν μήνυμα. Ερωτήματα που αφορούν τη χρήση της εργασίας για κερδοσκοπικό σκοπό πρέπει να απευθύνονται προς τον συγγραφέα.

Οι απόψεις και τα συμπεράσματα που περιέχονται σε αυτό το έγγραφο εκφράζουν το συγγραφέα και δεν πρέπει να ερμηνευθεί ότι αντιπροσωπεύουν τις επίσημες θέσεις του Εθνικού Μετσόβιου Πολυτεχνείου.

# Περίληψη

Οι Κβαντικοί Υπολογιστές είναι μια ιδέα που πρωτοπροτάθηκε από τους Paul Benioff και Richard Feynman τη δεκαετία του 80. Λόγω της εξαιρετικής δυσκολίας στην κατασκευή ενός σταθερού Κβαντικού Υπολογιστή για χρόνια η συζήτηση γύρω από τους Κβαντικούς Υπολογιστές έμενε σε θεωρητικό και αλγοριθμικό επίπεδο.

Αλγόριθμοι όπως του Shor και του Grover έδειχναν δεκαετίες πριν την ανωτερότητα ενός Κβαντικού μοντέλου σε σχέση με το υπάρχον Κλασσικό μοντέλο.

Τα τελευταία χρόνια με την εξέλιξη της τεχνολογίας μας δόθηκε η δυνατότητα να αναπτύξουμε ικανά και σταθερά Κβαντικά Υπολογιστικά Συστήματα, όπως Google (Quantum Supremacy), Microsoft, IBM Q, Rigetti, Honeywell and IonQ πολλά από τα οποία είναι ανοιχτά σε χρήση από το ευρύ κοινό, όπως τα σύγχρονα IAAS.

Σε αυτή την εργασία κάνουμε μια εισαγωγική μελέτη στους Κβαντικούς Αλγορίθμους και στη λογική πίσω από τον προγραμματισμό σε Κβαντικό Υπολογιστή. Μελετάμε αλγορίθμους Μηχανικής Μάθησης, πάνω σε Κβαντική λογική ενώ κατασκευάζουμε και έναν υβριδικό Κβαντικό-Κλασσικό kmeans για clustering τον οποίο τεστάρουμε σε περιβάλλοντα για Quantum Simulation και αποδεικνύουμε θεωρητικά την αποτελεσματικότητά του σε σχέση με τον Κλασσικό Αλγόριθμο σε επίπεδα χρονικής πολυπλοκότητας.

Τέλος ελέγχουμε την υλοποίησή μας σε πραγματικό περιβάλλον Κβαντικού Υπολογιστή στον IBMQ, εκθέτουμε τα αποτελέσματα και βγάζουμε συμπεράσματα ως προς τη σταθερότητα και το επίπεδο που βρίσκονται ακόμα σήμερα τα Κβαντικά Υπολογιστικά Συστήματα.

**Λέξεις Κλειδιά:** Κβαντικοί Υπολογιστές, Κβαντικοί Αλγόριθμοι, Μηχανική Μάθηση, kmeans, IBMQ, Qiskit



# Abstract

Quantum Computing is an idea that emerged from Paul Benioff and Richard Feynman from the early 1980s. Due to the lack of the technological means at the time for the construction of a stable Quantum Computer, for years Quantum Computing was only comprehended at a theoretical level and algorithmic level.

Algorithms like Shor's and Grover's demonstrated the supremacy of a real Quantum Model over the Classical Model.

Over the past years the technological progress has given us the tools to build fully stable and capable Quantum Computing Systems such as Google (Quantum Supremacy), Microsoft, IBM Q, Rigetti, Honeywell and IonQ. Many of them are accessible for individual users, like any other IAAS.

In this work we are introducing the Quantum Computing logic and take a deep look at some fundamental Quantum Algorithms, to understand the programming patterns of using Quantum Hardware. We then proceed to study some Quantum Machine Learning Algorithms. We built a hybrid of Quantum-Classical algorithm for kmeans clustering, which we examine and prove theoretically its supremacy over the classical one in means of execution time. We test its efficiency in a Quantum Simulation Environment.

Finally we are testing it on a real Quantum Hardware provided to us by IBMQ, and extract conclusions about the stability and efficiency of today's Quantum Computing Systems.

**Keywords:** Quantum Computing, Quantum Algorithms, Machine Learning, kmeans, IBMQ, Qiskit





# Ευχαριστίες

Με το πέρας αυτής της εργασίας, θα ήθελα να ευχαριστήσω τον καθηγητή μου Νεκτάριο Κοζύρη, που μου έδωσε την ευκαιρία να καταπιαστώ με ένα τόσο ενδιαφέρον θέμα, για την αμέριστη στήριξή του και εμπιστοσύνη του.

Θα ήθελα επίσης να ευχαριστήσω τον επιβλέποντά μου Κωνσταντίνο Νίκα για την καθοδήγησή του, την επικοινωνία μας πάνω στη δουλειά και την ενθάρρυνσή του στην έρευνά μου.

Αυτή η εργασία εκπονήθηκε πάνω σε δύσκολους προσωπικά καιρούς και εύκολο θα ήταν να έχει εγκαταληφθεί χωρίς την υποστήριξη των φίλων και της οικογένειάς μου, τους οποίους υπερευχαριστώ. Ευχαριστώ τον Νίκο Προβατά που με προέτρεψε και μου έδωσε κίνητρο να συνεχίσω αυτή την έρευνα σε καιρούς αρκετά πιεστικούς για εμένα.

Αυτή τη εργασία είναι αφιερωμένη στον πατέρα μου Παναιγώτη Μπιτσάκο, στη μητέρα μου Γεωργία Κομιανού και τον αδελφό μου Αλέξανδρο Μπιτσάκο.



# Contents

<b>Contents</b>	<b>11</b>
<b>List of Figures</b>	<b>13</b>
<b>1 Εισαγωγή</b>	<b>15</b>
1.1 Οι απαρχές της Κβαντικής Μηχανικής . . . . .	15
1.2 Κβαντικοί Υπολογιστές - η Αρχή . . . . .	18
<b>2 Μια εισαγωγή στην λογική του Quantum Computing</b>	<b>21</b>
2.1 Εισαγωγή στα Qubits . . . . .	21
2.1.1 Η Μαθηματική Αναπαράσταση ενός Qubit . . . . .	21
2.1.2 Tensor Product . . . . .	25
2.1.3 Quantum Entanglement . . . . .	26
<b>3 Λογικές Πύλες και Κυκλώματα Κβαντικών Υπολογιστών</b>	<b>29</b>
3.1 Βασικές Αρχές Κβαντικών Λογικών Πυλών . . . . .	29
3.2 Κάποιες Βασικές Κβαντικές Πύλες . . . . .	30
3.2.1 Identity Gate . . . . .	30
3.2.2 Not Gate . . . . .	30
3.2.3 Hadamard Gate . . . . .	30
3.2.4 Swap Gate . . . . .	30
3.2.5 Κβαντικά Κυκλώματα . . . . .	31
<b>4 Κβαντικοί Αλγόριθμοι - Κβαντικοί Αλγόριθμοι Μηχανικής Μάθησης</b>	<b>33</b>
4.1 Ο Αλγόριθμος Αναζήτησης του Grover . . . . .	34
4.2 Quantum Fourier Transformation . . . . .	38
4.3 A Quantum kmeans approach . . . . .	39
4.3.1 Swap Test . . . . .	41
4.3.2 Κβαντική αναπαράσταση ενός Κλασσικού state . . . . .	41
<b>5 Qiskit, IBM Quantum Experience - Υλοποίηση σε πραγματικό Κβαντικό Υπολογιστή και πειράματα</b>	<b>43</b>
5.1 Qiskit . . . . .	43
5.2 Quantum Discrete Fourier Transformation on Qiskit . . . . .	44
5.3 A Quantum kmeans on Qiskit . . . . .	47
<b>6 Επίλογος-Συμπεράσματα</b>	<b>55</b>
6.1 Συμπεράσματα . . . . .	55
6.2 Future work . . . . .	56



# List of Figures

1.1	Πείραμα μιας σχισμής . . . . .	16
1.2	Πείραμα δύο σχισμών με Παρατηρητή . . . . .	16
1.3	Πείραμα δύο σχισμών χωρίς Παρατηρητή . . . . .	17
1.4	. . . . .	17
2.1	. . . . .	22
2.2	Bloch Sphere . . . . .	22
3.1	Example of a Quantum Circuit . . . . .	31
3.2	Example of a Quantum Circuit 2 . . . . .	31
4.1	BQP . . . . .	34
4.2	Grover's elements list . . . . .	35
4.3	Grover's o cards deck . . . . .	35
4.4	A simple kmeans clustering example . . . . .	40
5.1	Qiskit Code 1 . . . . .	44
5.2	Qiskit Code 2 . . . . .	44
5.3	Qiskit Code 3 . . . . .	45
5.4	Qiskit QFT code . . . . .	46
5.5	Qiskit QFT circuit for 4 Qubits . . . . .	46
5.6	Result of QFT . . . . .	47
5.7	Code of clasical DFT . . . . .	47
5.8	Portion of Code of Quantum kmeans . . . . .	48
5.9	Quantum distance of 2 qubits circuit . . . . .	48
5.10	Ομοιότητα 2 states αναλόγως του αριθμού των shots . . . . .	49
5.11	Ομαδοποίηση 20 σημείων προϊόντος του χρόνου . . . . .	50
5.12	Ομαδοποίηση 1000 σημείων προϊόντος του χρόνου . . . . .	51
5.13	Final Dataset . . . . .	52
5.14	Pearson correlation . . . . .	52
5.15	Spearman correlation . . . . .	52
5.16	Features - Ctime and Qtime . . . . .	53
5.17	Shots and Qtime . . . . .	53



# 1

## Εισαγωγή

*Everything we call real is made of things that cannot be regarded as real  
If quantum mechanics hasn't profoundly shocked you, you haven't understood  
it yet.*

*Neils Bohr*

### 1.1 Οι απαρχές της Κβαντικής Μηχανικής

Ως τα τέλη του 19ου αιώνα η ανθρωπότητα έχοντας φτάσει στα όρια της κλασσικής σκέψης και ερευνήσει σε βάθος κάθε δυνατή πτυχή της, έφτανε στην πεποίθηση πως αργά ή σύντομα κάθε τι μικρό ή μεγάλο, πολύπλοκο ή απλό σε αυτό τον κόσμο θα μπορούσε σύντομα να αποκωδικοποιηθεί, να αναλυθεί και να κατανοηθεί από τον ανθρώπινο νου. Ετούτο αποτυπωνόταν και στις τέχνες μας, όπου κυριαρχούσε η δομή, η λογική και η συνέπεια.

Ίσως ο πρώτος φιλόσοφος που αναφέρθηκε στην τυχαιότητα και την απροσδιοριστία πάνω στην οποία δομείται ό,τι θεωρούμε ως αληθινό ή πραγματικό, ήταν ο Νίτσε όταν στο Τάδε Έφη Ζαρατούστρα έλεγε συμβολικά “Βλέπω σε ένα γιγάντιο κοσμικό τραπέζι Θεούς να ρίχνουν ζάρια που αποφασίζουν ό,τι μας δίνεται εμάς σαν αντανάκλαση ζωής”

Στις αρχές του 20ου αιώνα τρία σπουδαία θεωρήματα κλόνισαν συνθέμελα και μια για πάντα τις πεποιθήσεις μας για τον κόσμο που μας περιβάλλει, για το τι θεωρούμε ως πραγματικό και σταθερό και το ποια είναι η θέση μας μπροστά στη Δημιουργία. Το Θεώρημα της μη Πληρότητας του Godel [1] που απέδειξε την αναπόδραστη κλειστότητα της ανθρώπινης σκέψης και αντίληψης, η Ειδική και η Γενική Θεωρία της Σχετικότητας του Einstein [2] που απέδειξε πως έννοιες που θεωρούσαμε συμπαντικές σταθερές όπως ο χρόνος και ο χώρος, μόνο σταθερά δεν είναι και μεταβάλλονται αναλόγως της Μάζας και της Ενέργειας των αντικειμένων που περιβάλλουν αλλά και συσχετιστικά με τον παρατηρητή που τα αντιλαμβάνεται και τα μετράει και τέλος η Άννιση της Κβαντικής Μηχανικής που απέδειξε την απροσδιοριστία και την τυχαιότητα πάνω στην οποία δομούνται τα δομικά στοιχεία της ύπαρξης, όπως και την πρωτοφανή ιδέα πως το υποκείμενο της παρατήρησης έχει επιρροή πάνω στο παρατηρούμενο αντικείμενο.

Το περίφημο πείραμα των Σχισμών [3] μαρτυρά του λόγο του αληθές. Σε αυτό το πείραμα σωματίδια (ηλεκτρόνια ή φωτόνια) εκτοξεύονται προς έναν τοίχο. Εάν αυτός ο τοίχος έχει μια σχισμή τότε παρατηρούμε πίσω από τον τοίχο να δημιουργείται εν τέλει μια πυκνή γραμμή από σωματίδια.

Όταν αντί για μία, υπήρχαν δύο σχισμές στον τοίχο, τότε βλέπαμε πίσω του να δημιουργούνται όπως είναι φυσικό δύο πυκνές γραμμές από σωματίδια.

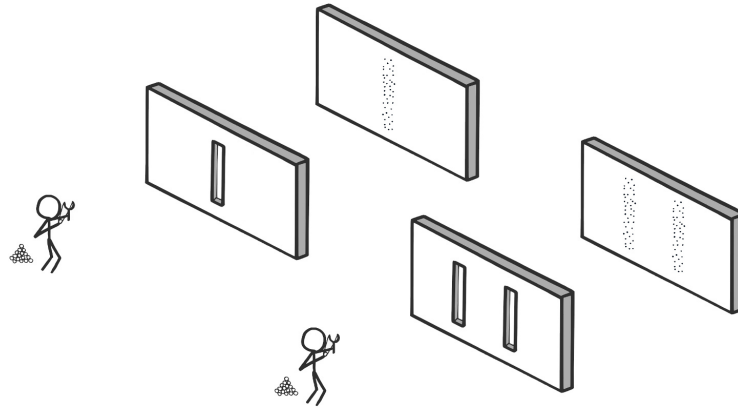


Figure 1.1: Πείραμα μιας σχισμής

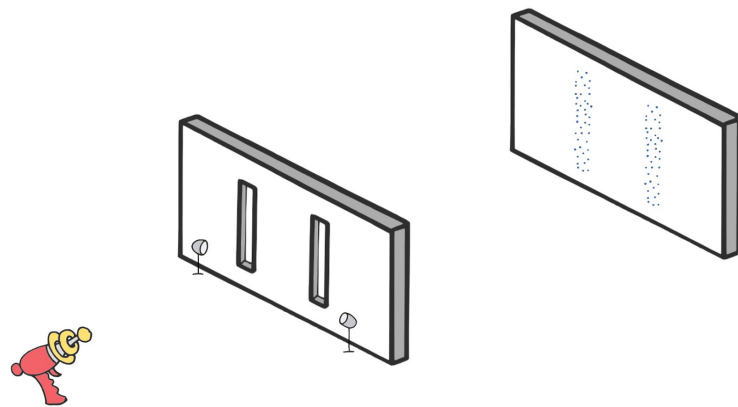


Figure 1.2: Πείραμα δύο σχισμών με Παρατηρητή



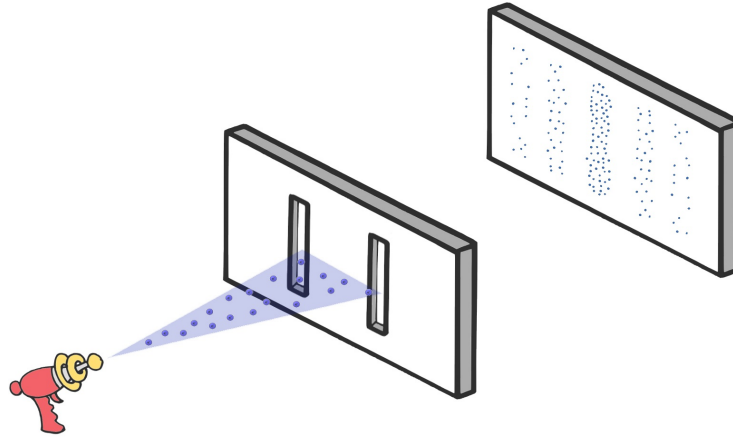


Figure 1.3: Πείραμα δύο σχισμών χωρίς Παρατηρητή

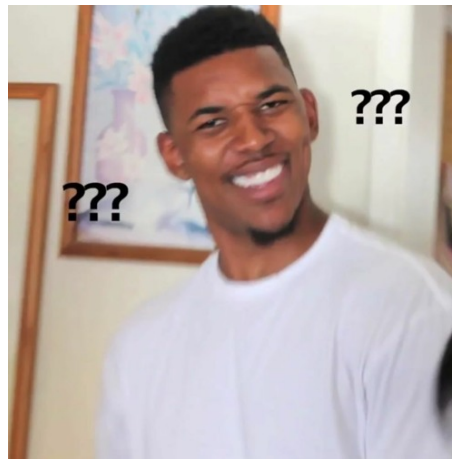


Figure 1.4

Όταν όμως επαναλαμβάνουμε το πείραμα χωρίς παρατηρητή, τότε κάτι ανεξήγητο και αξιωματικό γινόταν. Τότε τα σωματίδια πίσω από τον τοίχο κατέληγαν σε θέσεις, αρχικά φαινομενικά τυχαίες, που μελετώντας το μοτίβο τους, φαινόταν πως πρόκειται για μια πιθανοτική κυματική εξίσωση, την οποία περιέγραψε ο Schrodinger:

$$i\hbar \frac{\partial}{\partial t} \Psi(\mathbf{r}, t) = \hat{H} \Psi(\mathbf{r}, t)$$

Που σημαίνει πως τα σωματίδια χωρίς την ύπαρξη παρατηρητή δεν είχαν σταθερή σωματιδιακή υπόσταση αλλά ακολουθούσαν την πορεία ενός κύματος πιθανοτήτων που σταθερή ροπή και θέση θα έπαιρναν μόνο αν κάποιος παρατηρούσε την κίνησή τους.

Τότε δημιουργήθηκε η περίφημη σχολή της Κοπεγχάγης από αμειγώς κβαντικούς φυσικούς που προσπαθούσαν να απαντήσουν στο ερώτημα. Ο Κόσμος υπάρχει αυτούσιος ή επειδή υπάρχει νοήμονα ζωή για να τον κατανοήσει; Και αυτή η κατανόηση μήπως με τη σειρά της είναι μια πράξη πρακτικής Δημιουργίας;

## 1.2 Κβαντικοί Υπολογιστές - η Αρχή

Όπως καταλαβαίνετε γρήγορα ένα νέο επιστημονικό πεδίο αυτό της Κβαντικής Μηχανικής [4] αναπτύχθηκε με σπουδαίους επιστήμονες όπως οι Niels Bohr, Louis de Broglie, Albert Einstein, Erwin Schrödinger, Werner Karl Heisenberg, Von Neumann να συμμετέχουν είτε στο θεωρητικό είτε στο μαθηματικό υπόβαθρό της.

Στις αρχές του 1980 από ομιλίες των Richard Feynman, Yuri Manin, Paul Benioff προτάθηκε η ιδέα κατασκευής ενός Κβαντικού Υπολογιστή [5] στα πρότυπα του Κλασσικού Μοντέλου του Alan Turing.

Για χρόνια η ιδέα ενός τέτοιου υπολογιστή έμεινε στο θεωρητικό σχεδιασμό της ενώ δειλά δειλά άρχιζαν να κατασκευάζονται αλγόριθμοι που θα μπορούσαν να εκμεταλλευτούν έναν τέτοιο υπολογιστή. Ο πρώτος αλγόριθμος που παρουσίασε τη σαφή υπεροχή ενός Κβαντικού Υπολογιστή απέναντι στο κλασσικό μοντέλο ήταν το Αλγόριθμος του Shor [6], το 1994, για το factorizing μεγάλων Πρώτων Αριθμών. Θεωρητικά αυτός ο Αλγόριθμος σε ισχύ δύναται να σπάσει τα περισσότερα κρυπτοσυστήματα που χρησιμοποιούνται σήμερα.

Το 1996 ο Lov Grover [7] παρουσίασε έναν Κβαντικό Αλγόριθμο αναζήτησης για βάσεις δεδομένων που έδινε μια πολυωνυμική βελτίωση στους αλγορίθμους αναζήτησης που χρησιμοποιούμε σήμερα. Θα παρουσιάσουμε αυτό τον αλγόριθμο στη συνέχεια.

Το 1998, παρουσιάζεται ένας Κβαντικός Υπολογιστής που δουλεύει με 2 qubits (το αντίστοιχο των κλασσικών bits για τον Κβαντικό Υπολογιστή, όπως θα δούμε αργότερα). Αυτός ο υπολογιστής χρησιμοποιήθηκε για να εξετάσει τον αλγόριθμο του Grover με επιτυχία.

Το 2000, ο πρώτος Κβαντικός Υπολογιστής 5 qubit από το Πολυτεχνείο του Μονάχου και ο πρώτος Κβαντικός Υπολογιστής 7 qubit από το Los Alamos National Laboratory παρουσιάστηκαν με επιτυχία, ενώ εκδόθηκε το κλασσικό πλέον Quantum Computation and Quantum Information.

Το 2001 στο εργαστήριο της IBM παρουσιάστηκε για πρώτη φορά επιτυχώς η εκτέλεση του αλγορίθμου του Shor. Το 2005 για πρώτη φορά επιτυχώς είχαμε μεταφορά πληροφορίας μεταξύ δύο Κβαντικών Μνημών σε συνεργασία του Harvard University και του Georgia Institute of Technology. Το 2011 κατασκευάστηκε ο πρώτος καταχωρητής 14ων qubits. Το 2012 κατασκευάστηκε ένας προσωμοιωτής 300ων qubits ενώ ιδρύθηκε η 1QB Information Technologies (1QBit), η πρώτη εταιρία λογισμικού με αποκλειστική ειδικευση τους Κβαντικούς Υπολογιστές.

Το 2015, η πρώτη λογική πύλη πυριτίου για 2 qubits κατασκευάζεται. Το 2016 η IBM παρουσιάζει το Quantum Experience ένα διαδυκτιακό API ανοιχτό στο κοινό για χρήση Κβαντικών Υπολογιστικών Συστημάτων (θα το χρησιμοποιήσουμε αργότερα σε ετούτη την εργασία), ενώ το 2017 παρουσιάζει έναν Κβαντικό Υπολογιστή 17 qubits με ολοκληρωμένο API για benchmarking

Το 2019 η IBM κατασκευάζει τον μεγαλύτερο έως σήμερα Κβαντικό Υπολογιστή της, των 53ών qubits, ενώ η Google ανακοινώνει τον ισχυρισμό πως πέτυχε το Quantum Supremacy [8] χρησιμοποιώντας τον Κβαντικό Υπολογιστή της για να λύσει σε 2.5 μέρες έναν υπολογισμό που ένας Κλασσικός Υπολογιστής θα χρειαζόταν τουλάχιστον 10000 χρόνια για να λύσει. Ο ισχυρισμός αυτός έχει δεχθεί έντονη κριτική για τη βασιμότητά του έκτοτε.

Από το 2020 έχουν κατασκευαστεί και τα πρώτα κατανεμημένα συστήματα Κβαντικών Υπολογιστών. Σήμερα πολλές εταιρίες και εκπαιδευτικά ιδρύματα προσφέρουν λύσεις Κβαντικών συστημάτων, όπως οι Google (Quantum Supremacy), Microsoft, IBM Q, Rigetti, Honeywell and IonQ αν και είμαστε ακόμα μακριά από έναν Κβαντικό Υπολογιστή, ανοιχτό στο κοινό με σταθερή απόδοση και ανωτερότητα από ένα κλασσικό μοντέλο καθώς απέχουν ο θεωρητικός σχεδιασμός από τον πρακτικό σχεδιασμό ενός υπολογιστή που στο υποατομικό επίπεδο λειτουργεί με Κβαντική Λογική, από τα πιο απλά προβλήματα όπως τη δυσκολία

του να θέσεις σωματίδια στην κατάλληλη κατάσταση για λειτουργία σε Κβαντική Λογική (αποθηκευτικοί λόγοι, θερμοκρασία, κόστος, αποθήκευση) όπως και γιατί τα αποτελέσματα που μας δίνει ένας Κβαντικός Υπολογιστής δεν είναι ντετερμινιστικά αλλά πιθανοτικά, που σημαίνει πως πρέπει να επαναληφθεί πολλές φορές μια εκτέλεση μέχρι να βγάλουμε ασφαλή αποτελέσματα.



## 2

# Μια εισαγωγή στην λογική του Quantum Computing

*We must be clear that when it comes to atoms, language can be used only as in poetry.*

*Neils Bohr*

## 2.1 Εισαγωγή στα Qubits

Θα αρχίσουμε το ταξίδι μας στην λογική ενός Κβαντικού Υπολογιστή από το δομικό στοιχείο κάθε Κβαντικού Υπολογιστή τα Qubits [9]. Τα qubits όπως και τα κλασσικά bits αποτελούνται από σωματίδια (πχ ηλεκτρόνια) και αποτελούν την πιο μικρή μονάδα πληροφορίας ενός Κβαντικού Υπολογιστή.

Μπορούν να υπάρξουν σταθερά σε δύο καταστάσεις. Παραδείγματος χάριν το spin [10] ενός ηλεκτρονίου όπου μπορούμε να χρησιμοποιήσουμε σαν τις δύο καταστάσεις το spin down και το spin up. Η διαφορά με το 0 και 1 των κλασσικών bits είναι πως στα qubits πριν τη μέτρηση το σωματίδιο δε βρίσκεται δυαδικά σε μια από τις δυο καταστάσεις αλλά υπάρχει σαν κύμα πιθανοτήτων, μια ιδιότητα που ονομάζεται superposition [11].

Οπτικά μπορούμε να μοντελοποιήσουμε το κύμα πιθανοτήτων της κατάστασης του σωματιδίου στην Bloch Sphere [12] που βλέπουμε στην εικόνα 2.2

Παρατηρούμε πως το σωματίδιο πριν μετρηθεί και καταρρεύσει σε μία από τις καταστάσεις 0 ή 1 μπορεί πιθανοτικά να βρίσκεται σε πολλές άλλες καταστάσεις που εκτείνονται πάνω στην τρισδιάστατη σφαίρα στον μιγαδικό χώρο.

Τα προτερήματα είναι τεράστια και όπως θα γίνει αντιληπτό κατά την ανάγνωση ενώ σε έναν Κλασσικό Υπολογιστή με  $n$  bits μπορούμε να αναπαραστήσουμε πληροφορία μεγέθους το πολύ  $n$ , στον αντίστοιχο Κβαντικό Υπολογιστή με  $n$  qubits μπορούμε να αναπαραστήσουμε πληροφορία που χρειάζεται  $2^n$  μιγαδικούς αριθμούς.

### 2.1.1 Η Μαθηματική Αναπαράσταση ενός Qubit

Ένα qubit αναπαρίσταται σαν συνδιασμός δύο καταστάσεων, του 0 και του 1. Χρησιμοποιώντας την dirac notation έχουμε

$$|\Psi\rangle = \alpha |0\rangle + \beta |1\rangle$$

όπου τα  $\alpha$ ,  $\beta$  αποκαλούμε amplitudes της κάθε κατάστασης.

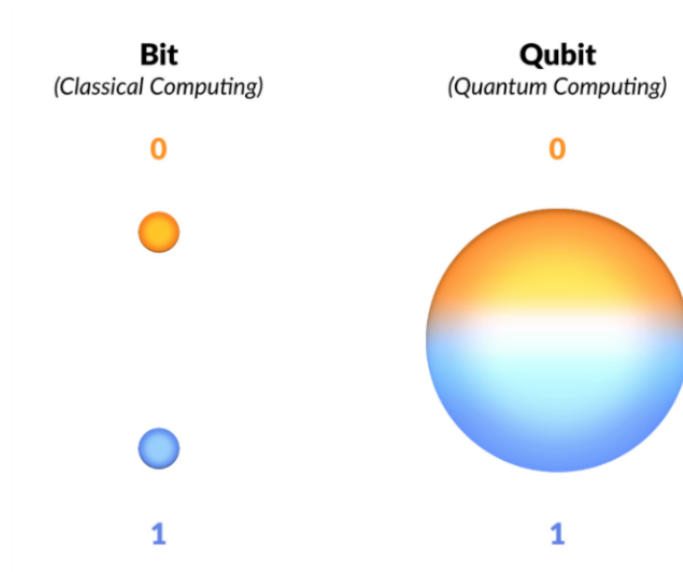


Figure 2.1

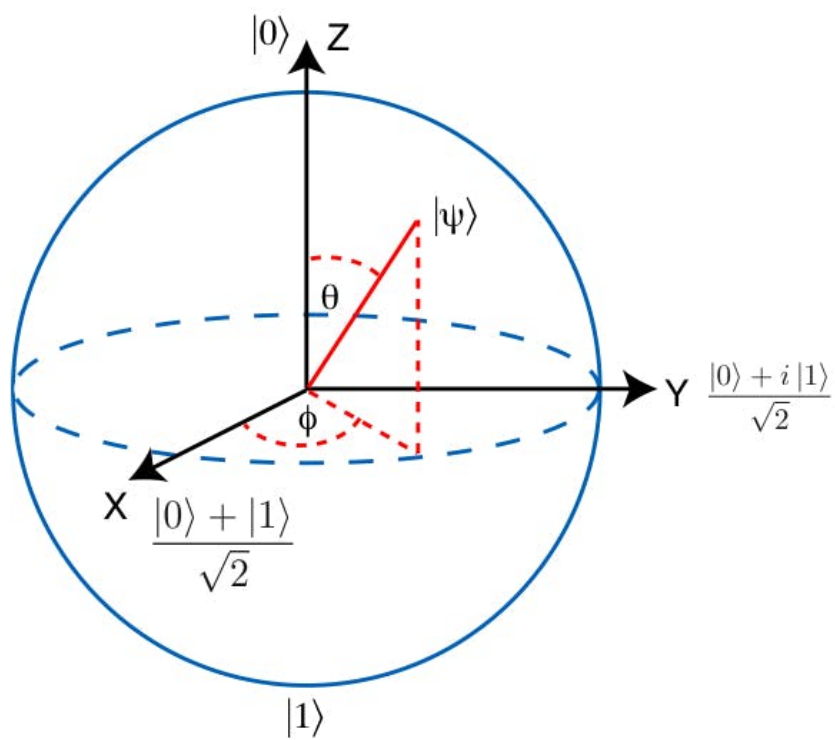


Figure 2.2: Bloch Sphere

$$\langle 0| = [1 \quad 0] |0\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

Dirac Notation

$$\langle 1| = [0 \quad 1] |1\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

Dirac Notation

$$\langle +| = \frac{1}{\sqrt{2}} |0\rangle + \frac{1}{\sqrt{2}} |1\rangle$$

$$\langle -| = \frac{1}{\sqrt{2}} |0\rangle - \frac{1}{\sqrt{2}} |1\rangle$$

$$\langle i| = \frac{1}{\sqrt{2}} |0\rangle + \frac{i}{\sqrt{2}} |1\rangle$$

$$\langle -i| = \frac{1}{\sqrt{2}} |0\rangle - \frac{i}{\sqrt{2}} |1\rangle$$

Fundamental Qubit states

Τα σύμβολα  $\langle |, | \rangle$  [13] ονομάζονται αντίστοιχα bra, ket και αναπαριστούν έναν πίνακα σε οριζόντια ή κάθετη μορφή.

To Bra notation:  $\langle a|$  είναι ένα conjugate transpose του ket notation:  $|a\rangle$

Ενώ το Bracket  $\langle \phi | \psi \rangle$  αναπαριστά εσωτερικό γινόμενο.

Παρατηρούμε πως στις θέσεις του πίνακα μπαίνουν τα Amplitudes δηλαδή όπως θα δούμε σε λίγο η πιθανότητα ένα qubit να παίρνει την τιμή 1 ή 0 αντίστοιχα.

Για να βρούμε αυτή την πιθανότητα πρέπει να υπολογίσουμε το **Conjugate Transpose του Amplitude, που πρέπει να τονιστεί ξανά πως δεν είναι το απόλυτο του βάρους, αλλά το Conjugate Transpose.**

Τουτέστιν, έστω ότι

$$A = \begin{bmatrix} 1 & -2-i & 5 \\ 1+i & i & 4-2i \end{bmatrix}$$

To transpose:

$$A^T = \begin{bmatrix} 1 & 1+i \\ -2-i & i \\ 5 & 4-2i \end{bmatrix}$$

To conjugate transpose:

$$A^H = \begin{bmatrix} 1 & 1-i \\ -2+i & -i \\ 5 & 4+2i \end{bmatrix}$$

Εάν έχουμε πάνω από ένα qubit τότε μιλάμε για ένα σύστημα qubits που μπορεί να καταρρεύσει μόνο σε μία θέση. Έτσι πχ μπορεί να έχουμε το

$$\langle \psi| = [0 \quad 0 \quad \dots \quad 1 \quad \dots \quad 0]^T$$

Ενώ πριν τη μέτρηση και πριν το σύστημα καταρρεύσει τότε η κατάσταση του περιγράφεται από μια εξίσωση της μορφής

$$\langle \psi | = [c_0 \quad c_1 \quad \dots \quad c_n]^T$$

Όπως είναι λογικό το άθροισμα των Conjugate Transposes των Amplitudes πρέπει να ισούται με 1, οπότε έχουμε

$$|c_1|^2 + |c_2|^2 + \dots + |c_n|^2 = \sum_{n=1}^n |c_n|^2 = 1$$

Όπου για την παρακάτω κατάσταση

$$\langle \Psi | = c_1 |x_1\rangle + c_2 |x_2\rangle = \frac{i}{2} \begin{bmatrix} 1 \\ 0 \end{bmatrix} + \frac{\sqrt{3}}{2} \begin{bmatrix} 0 \\ 1 \end{bmatrix} = \begin{bmatrix} \frac{i}{2} \\ \frac{\sqrt{3}}{2} \end{bmatrix}$$

έχουμε την πιθανότητα το σύστημα να καταρρεύσει στη θέση  $|x_1\rangle$  από το square norm του μιγαδικού  $c_1$ :

$$|c_1|^2 = c_1^* c_1 = \frac{-i}{2} \frac{i}{2} = \frac{-i^2}{2^2} = \frac{1}{4}$$



Σημαντικό στοιχείο στο Κβαντικό Μοντέλο όπως έχουμε ήδη καταλάβει είναι η μέτρηση και η πιθανότητα ένα σύστημα να βρεθεί στην θέση  $\chi$  ή στη θέση  $\psi$ .

Λέμε λοιπόν, πως η πιθανότητα να βρούμε ένα σύστημα  $|\psi\rangle$  σε μια κατάσταση  $|\phi\rangle$  δίνεται από τον τύπο

$$|\langle\phi|\psi\rangle|^2$$

Οπότε στο προηγούμενο παράδειγμα μπορούμε να βρούμε την πιθανότητα το σύστημα να καταρρεύσει στο  $|\chi_1\rangle$  ακολούθως:

$$\begin{aligned}\langle x_1|\psi\rangle &= \langle x_1|c_1|x_1\rangle + \langle x_1|c_2|x_2\rangle = c_1\langle x_1|x_1\rangle + c_2\langle x_1|x_2\rangle \\ &= \frac{i}{2} \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix} + \frac{\sqrt{3}}{2} \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \end{bmatrix} = \frac{i}{2}\end{aligned}$$

Όπου το Square Norm γίνεται

$$|\langle\phi|\psi\rangle|^2 = |\langle\phi|\psi\rangle|^*|\langle\phi|\psi\rangle| = \frac{-i}{2}\frac{i}{2} = \frac{1}{4}$$

### 2.1.2 Tensor Product

Πριν προχωρήσουμε στην αναπαράσταση πολλών qubits μαζί, πρέπει να γνωρίσουμε την έννοια του tensor product, έναν τύπο γινομένου μεταξύ δύο πινάκων  $V, W$  που ματσάρει κάθε ζεύγος  $(u, w), u \in V, w \in W$  σε ένα στοιχείο  $V \otimes W$ .

Μπορούμε να το δούμε σχηματικά στις ακόλουθες εξισώσεις

$$|0\rangle \otimes |1\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \otimes \begin{bmatrix} 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & * & \begin{bmatrix} 0 \\ 1 \end{bmatrix} \\ 0 & * & \begin{bmatrix} 0 \\ 1 \end{bmatrix} \end{bmatrix} = \begin{bmatrix} 1 & * & 0 \\ 1 & * & 1 \\ 0 & * & 0 \\ 0 & * & 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix}$$

Το tensor product μας επιτρέπει να δημιουργούμε συνδιασμούς πολλών qubits καταλήγοντας σε μεγάλα mixture Quantum states. Λόγου χάρη αν έχουμε δύο qubits μπορούμε να βρούμε το state βρίσκοντας το tensor product τους.

Παραδείγματος χάριν

$$\begin{aligned}|\Psi\rangle &= |\psi_\chi\rangle \otimes |\psi_\psi\rangle \\ &= (c_{1,\chi}|\chi_1\rangle + (c_{2,\chi}|\chi_2\rangle)) \otimes (c_{1,\psi}|\psi_1\rangle + (c_{2,\psi}|\psi_2\rangle)) \\ &= c_1|\chi_1\psi_1\rangle + c_2|\chi_1\psi_2\rangle + c_3|\chi_2\psi_1\rangle + c_4|\chi_2\psi_2\rangle \\ &= c_1 \begin{bmatrix} 1 \\ 0 \end{bmatrix} \otimes \begin{bmatrix} 1 \\ 0 \end{bmatrix} + c_2 \begin{bmatrix} 1 \\ 0 \end{bmatrix} \otimes \begin{bmatrix} 0 \\ 1 \end{bmatrix} + c_3 \begin{bmatrix} 0 \\ 1 \end{bmatrix} \otimes \begin{bmatrix} 1 \\ 0 \end{bmatrix} + c_4 \begin{bmatrix} 0 \\ 1 \end{bmatrix} \otimes \begin{bmatrix} 0 \\ 1 \end{bmatrix} \\ &= c_1 \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} + c_2 \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} + c_3 \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix} + c_4 \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}\end{aligned}$$

### 2.1.3 Quantum Entanglement

Αφού έχουμε δει τις δύο σημαντικότερες έννοιες των Κβαντικών Υπολογιστών, ήτοι τη **Μέτρηση** [14]. και το **Superposition** ήρθε η ώρα να δούμε την ίσως πιο παράδοξη έννοια. Το **Quantum Entanglement** [15].

Στην Κβαντική Μηχανική το Quantum Entanglement ή αλλιώς Κβαντικός Εναγκαλισμός ονομάζεται το φαινόμενο όπου δύο σωματίδια, **ασχέτως σε ποιο σημείο του σύμπαντος βρίσκεται ο καθένα και πόσο απέχουν**, αλληλεπιδρούν μεταξύ τους και οποιαδήποτε αλλαγή στην κατάσταση του ενός επιφέρει αλλαγή στην κατάσταση του άλλου.

Το φαινόμενο αυτό καταρρίπτει κάθε γνωστό όριο της φύσης όπως την ταχύτητα του φωτός. Τα σωματίδια βρίσκονται μεταξύ τους σε "επαφή" όσο και να απέχουν το ένα από το άλλο, δηλαδή η πληροφορία μεταφέρεται από το ένα στο άλλο ακαριαία.

Στους Κβαντικούς Υπολογιστές παρατηρούμε πάλι αυτό το φαινόμενο. Άπαξ και φέρουμε δύο qubits σε Κβαντικό Εναγκαλισμό τότε οποιαδήποτε αλλαγή στο ένα θα επιφέρει αλλαγή και στο άλλο. Μπορούμε να δούμε σχηματικά αυτό το φαινόμενο στους παρακάτω πίνακες:

$$|\Psi\rangle = \frac{1}{2} |aa\rangle + \frac{1}{2} |ba\rangle + \frac{1}{2} |ab\rangle + \frac{1}{2} |bb\rangle$$

$$|\Phi\rangle = \frac{1}{\sqrt{2}} |aa\rangle + \frac{1}{\sqrt{2}} |bb\rangle$$

Possible outcome of state $ \psi\rangle$		
Particle 1: $ a\rangle$	$ \psi\rangle \Rightarrow  aa\rangle : \frac{1}{4}$	$ \psi\rangle \Rightarrow  ab\rangle : \frac{1}{4}$
Particle 2: $ b\rangle$	$ \psi\rangle \Rightarrow  ba\rangle : \frac{1}{4}$	$ \psi\rangle \Rightarrow  bb\rangle : \frac{1}{4}$

Possible outcome of state $ \phi\rangle$		
Particle 1: $ a\rangle$	$ \phi\rangle \Rightarrow  aa\rangle : \frac{1}{2}$	$ \phi\rangle \Rightarrow  ab\rangle : 0$
Particle 2: $ b\rangle$	$ \phi\rangle \Rightarrow  ba\rangle : 0$	$ \phi\rangle \Rightarrow  bb\rangle : \frac{1}{2}$

Τι παρατηρούμε στους παραπάνω πίνακες; Ενώ για το Superposition  $\Psi$  βλέπουμε πως λόγω χάρη υπάρχει πιθανότητα 50% να βρούμε το πρώτο qubit στη θέση a και τότε υπάρχει 25% πιθανότητα να βρούμε το δεύτερο qubit στη θέση a και αλλο τόσο να το βρούμε στη θέση b. Όλα φυσιολογικά ως εδώ.

Πάμε τώρα στο Superposition  $\Phi$ . Εδώ παρατηρούμε το εξής παράδοξο. Υπάρχει πάλι η πιθανότητα 50% να βρούμε το πρώτο qubit στη θέση a. Εάν όμως το πρώτο qubit είναι στη θέση a τότε το δεύτερο qubit θα βρίσκεται οπωσδήποτε στη θέση a νομοτελειακά και ποτέ στη θέση b !!

Ένα πραγματικά ακατανόητο παράδοξο της φύσης, το οποίο ως είθισται στο είδος μας, θα το κατακερματίσουμε και θα το χρησιμοποιήσουμε (χωρίς να το καταλαβαίνουμε) στη μοντελοποίηση και τη λύση των προβλημάτων μας στους Κβαντικούς Υπολογιστές!

Πριν προχωρήσουμε στο επόμενο κεφάλαιο, ας δούμε ένα παράδειγμα μιας Superposition 8 qubits, έχοντας πλέον την απαραίτητη οικιότητα:

$$|01010001\rangle = \begin{bmatrix} 0_{|00000000\rangle} \\ 0_{|00000001\rangle} \\ \cdot \\ \cdot \\ \cdot \\ 0_{|01010000\rangle} \\ 1_{|01010001\rangle} \\ \cdot \\ \cdot \\ \cdot \\ 0_{|11111110\rangle} \\ 0_{|11111111\rangle} \end{bmatrix}$$

Αφού

$$|\Psi\rangle = c_1 |00000000\rangle + ..c_{82} |01010001\rangle + .. + c_{255} |11111110\rangle + c_{256} |11111111\rangle = \sum_{i=1}^N c_i |i\rangle$$



### 3

## Λογικές Πύλες και Κυκλώματα Κβαντικών Υπολογιστών

*The 'paradox' is only a conflict between reality and your feeling of what reality  
'ought to be'*

*Richards Feynman*

### 3.1 Βασικές Αρχές Κβαντικών Λογικών Πυλών

Ο προγραμματισμός Κβαντικών Υπολογιστών διαφέρει άρδην από το αντίστοιχο Κλασσικό Μοντέλο. Στους Κβαντικούς υπολογιστές προγραμματίζουμε σε επίπεδο λογικών πυλών [16] Ή ακόμα καλύτερα κατασκευάζουμε Κβαντικά Κυκλώματα που διαχειρίζονται καταστάσεις από πολλά qubits σε Superposition με τη βοήθεια Λογικών Πυλών.

Οι Λογικές Πύλες των Κβαντικών Υπολογιστών διαφέρουν σημαντικά από τις αντίστοιχες του Κλασσικού Μοντέλου. Πιο συγκεκριμένα

- Πρέπει να διατηρούν το άθροισμα του square norm του αρχικού Superposition απαράλλαχτο, από την είσοδο στην έξοδο της πύλης
- Πρέπει να είναι αναστρέψιμες, reversible. Δηλαδή κάθε Superposition σύστημα, θα πρέπει να μπορεί να πάει πίσω αντίστροφα στην αρχική του κατάσταση αν του εφαρμοστεί ξανά η ίδια λογική πύλη

Κάθε Κβαντική πύλη αναπαριστάται από έναν τένσορα, έναν πίνακα  $n \times n$  που εφαρμόζεται σε ένα Κβαντικό state και αυτός ο τένσορας οφείλει να είναι **Unitary Matrix** για να ακολουθεί τις δύο παραπάνω ιδιότητες. Δηλαδή:

$$U^*U = UU^* = I$$

Για να εφαρμόσουμε μια Κβαντική Πύλη σε ένα σύστημα βρίσκουμε το tensor product της πύλης με το σύστημα.

Στο επόμενο κεφάλαιο θα δούμε κάποιες από τις πιο γνωστές Κβαντικές Πύλες και πώς μπορούμε να τις χρησιμοποιήσουμε.

## 3.2 Κάποιες Βασικές Κβαντικές Πύλες

### 3.2.1 Identity Gate

Η Identity Gate παίρνει ένα Σύστημα και στην έξοδό της δίνει το ίδιο Σύστημα απaráλλακτο.

$$I = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

### 3.2.2 Not Gate

Η πύλη Not παίρνει ένα σύστημα και αντιστρέφει όλα τα qubits του:

$$NOT = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$$

$$\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

$$\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

### 3.2.3 Hadamard Gate

Η πιο γνωστή και διαδομένη ίσως Κβαντική Πύλη είναι η Hadamard Gate. Παίρνει ένα σύστημα από superposition qubits και παράγει ένα αντίστοιχο σύστημα με ισοπίθανη κάθε κατάσταση του συστήματος της εισόδου.

$$H|0\rangle = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ 1 \end{bmatrix} = \frac{|0\rangle + |1\rangle}{\sqrt{2}}$$

$$H|1\rangle = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \end{bmatrix} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ -1 \end{bmatrix} = \frac{|0\rangle - |1\rangle}{\sqrt{2}}$$

### 3.2.4 Swap Gate

Η Swap Gate όπως μαρτυρά το όνομά της παίρνει σαν είσοδο ένα σύστημα από κάποια qubits και αλλάζει τη θέση τους.

$$SWAP = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$SWAP|01\rangle = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix} = |10\rangle$$

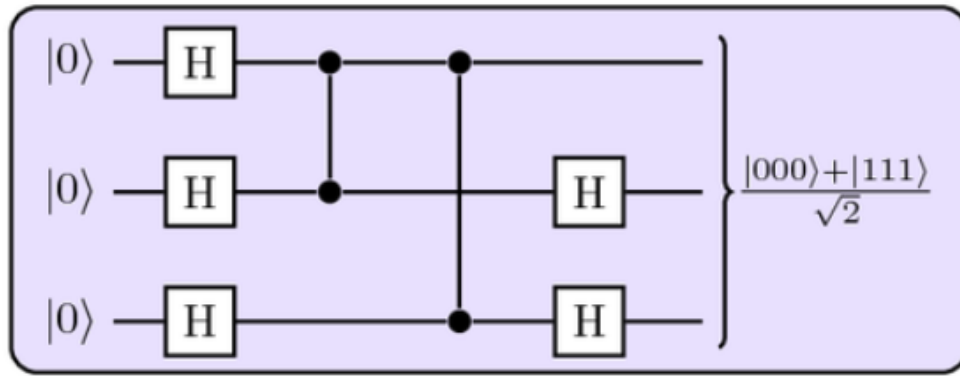


Figure 3.1: Example of a Quantum Circuit

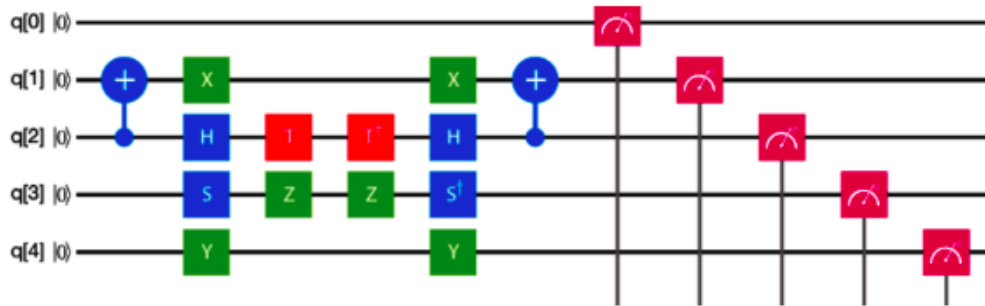


Figure 3.2: Example of a Quantum Circuit 2

### 3.2.5 Κβαντικά Κυκλώματα

Χρησιμοποιώντας τις Κβαντικές πύλες που γνωρίσαμε παραπάνω αλλά και άλλες όπως οι Toffoli, Pauli, Phase Shifting Gates, Rotation operator gates [17] κατασκευάζουμε Κβαντικά Κυκλώματα [18] που μανιπουλάρουν εισόδους από superposition πολλών qubits για να παράξουμε λογικά προγράμματα που τελούν επιθυμητές για εμάς λειτουργίες.

Σχήματα τέτοιων Κβαντικών Κυκλωμάτων βλέπουμε στις φωτογραφίες 3.1, 3.2:

Στο πρώτο παράδειγμα λ.χ. 3.1 βλέπουμε ένα Superposition τριών qubits, όπου αρχικά και τα τρία τα περνάμε από μια Hadamard Gate, ύστερα το δεύτερο και το τρίτο από αυτά από ένα Cnot Gate που κοντρολλάρεται από το πρώτο και τέλος, το δεύτερο και το τρίτο από μια Hadamard Gate.

Αποτέλεσμα από τρία qubits στην κατάσταση 0 έχουμε τρία Entagled qubits στην κατάσταση  $\frac{|000\rangle + |111\rangle}{\sqrt{2}}$ .





# Κβαντικοί Αλγόριθμοι - Κβαντικοί Αλγόριθμοι Μηχανικής Μάθησης

*I do not like it, and I am sorry I ever had anything to do with it*

*Erwin Schrödinger*

Έχοντας κάνει ένα μικρό ταξίδι στα προηγούμενα κεφάλαια μπορούμε τώρα να έχουμε μια σαφή εικόνα του πώς μπορεί να προγραμματίσει κανείς σε κβαντική λογική.

Τσως αυτή τη στιγμή ο αναγνώστης να θεωρήσει πως ο Κβαντικός Προγραμματισμός δεν είναι τίποτα άλλο από Λογική Σχεδίαση σε άλλο domain. Όχι όμως. Τώρα που κατανοήσαμε πως παίρνοντας κάποια qubits, μπορούμε με λογικές πύλες και κατ' επέκταση κυκλώματα να τα μανιπιουλάρουμε και να φτάσουμε σε επιθυμητά Superposition, τώρα είναι που γίνεται ενδιαφέρον.

Τώρα θα δούμε πως μπορούμε να χρησιμοποιήσουμε τη μαγεία της Μαθηματικής Λογικής του Κβαντικού Προγραμματισμού, ώστε χρησιμοποιώντας Κβαντικά Κυκλώματα να λύσουμε πρακτικά προβλήματα.

Αρχής γενομένης από τον Αλγόριθμο Αναζήτησης του Grover 4.1.

Πριν όμως προχωρήσουμε στον Αλγόριθμο, ας συζητήσουμε λίγο για την Κβαντική λογική της χρονικής πολυπλοκότητας. Αυτό είναι ένα όχι τόσο ξεκάθαρο σημείο, καθώς οι Κβαντικοί Αλγόριθμοι, είναι πρώτα από όλα Πιθανοτικοί. Δηλαδή η έξοδος μετά από μία εκτέλεση, δεν είναι καθόλου σίγουρο πως θα είναι σωστή. **Πρέπει η εκτέλεση να επαναληφθεί πολλές φορές μέχρι να οδηγηθούμε σε ασφαλές συμπέρασμα**

Αυτό προσθέτει αισθητά στην χρονική πολυπλοκότητα ενός Κβαντικού Αλγορίθμου. Ονομάζουμε τα προβλήματα που μπορούν να λυθούν με Κβαντικό Προγραμματισμό BQP (bounded error, quantum, polynomial time) [19]. Είναι η κλάση των προβλημάτων που μπορούν να λυθούν σε πολυωνυμικό χρόνο με πιθανότητα λάθους το πολύ  $1/3$ .

Η κλάση BQP (bounded error, quantum, polynomial time) είναι το κβαντικό ισοδύναμο της κλάσης BPP (bounded error, probabilistic, polynomial time) [20] των Κλασικών Υπολογιστών. Είναι αποδεδειγμένο πως  $BPP \subseteq BQP$  (bounded error, quantum, polynomial time) και εικόζεται χωρίς να έχει αποδειχθεί πως  $BQP \subseteq BPP$  (bounded error, quantum, polynomial time) *not*  $\subseteq BPP$  (bounded error, probabilistic, polynomial time).

Δεν γνωρίζουμε ακόμα την ακριβή θέση του BQP (bounded error, quantum, polynomial

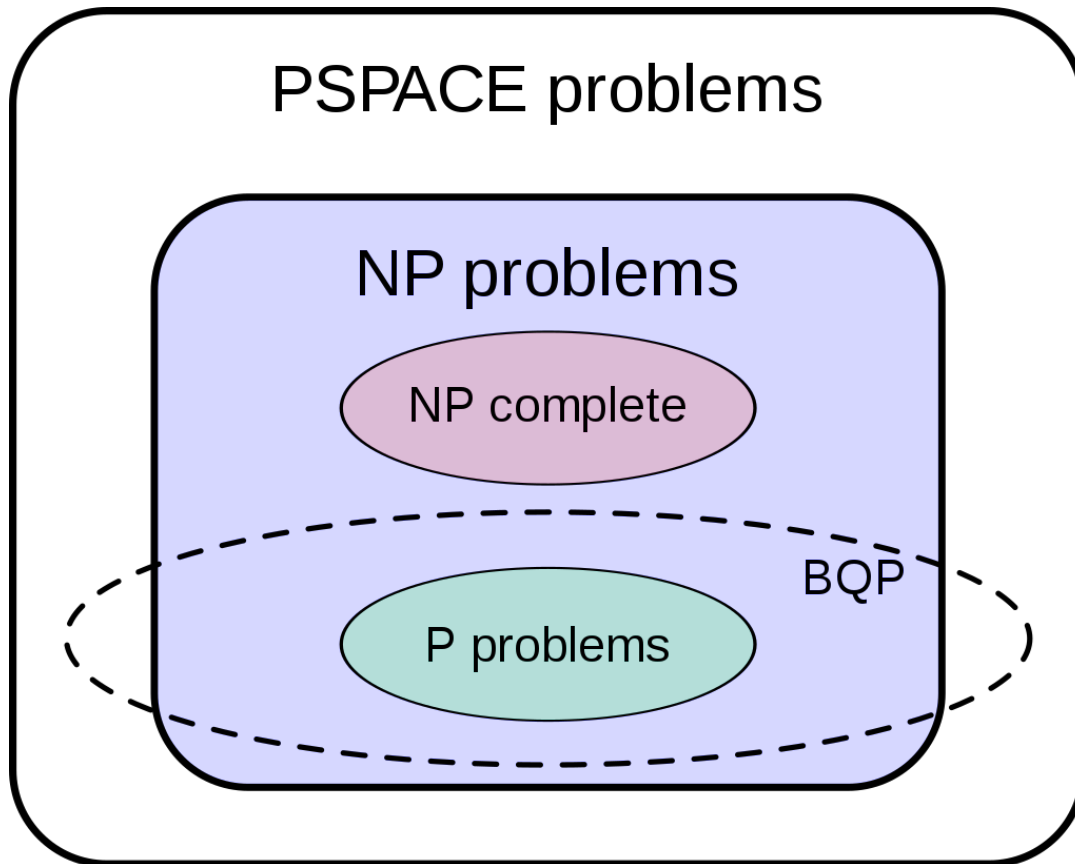


Figure 4.1: BQP

time) στο PSPACE [21] αλλά ξέρουμε πως  $P \subseteq BQP$  (bounded error, quantum, polynomial time)  $\subseteq PSPACE$ . Στην εικόνα 4.1 μπορούμε να δούμε πού περίπου εικάζεται πως βρίσκεται το BQP (bounded error, quantum, polynomial time) στο PSPACE.

#### 4.1 Ο Αλγόριθμος Αναζήτησης του Grover

Ο Αλγόριθμος του Grover πρωτοπαρουσιάστηκε το 1996 από τον Lov Grover και αποτελεί έναν Κβαντικό Αλγόριθμο Αναζήτησης που μπορεί με εξαιρετικά μεγάλη πιθανότητα να εντοπίσει ένα στοιχείο μέσα σε ένα σύνολο δεδομένων 4.2, 4.3 που μπαίνει ως είσοδος στον αλγόριθμο με χρονική πολυπλοκότητα  $O(\sqrt{n})$ . Σαν  $O(\sqrt{n})$  ορίζουμε το πόσες φορές θα πρέπει να εκτιμηθεί η τιμή της συνάρτησης που μπαίνει σαν “μαύρο κουτί” είσοδος στον αλγόριθμο.

Προφανώς ένα παρόμοιο πρόβλημα σε μη ταξινομημένο σύνολο δεδομένων μπορεί να επιλυθεί από έναν κλασσικό αλγόριθμο σε  $O(n)$

Ο Αλγόριθμος χρησιμοποιείται και από άλλους Κβαντικούς Αλγορίθμους για την επίλυση NP προβλημάτων, καθώς πολλοί Κβαντικοί Αλγόριθμοι (όπως και πολλοί κλασσικοί) χρησιμοποιούν αναζήτηση.

Ουσιαστικά αυτό που κάνει ο αλγόριθμος είναι να εφαρμόζει μια συνάρτηση σε ένα σύνολο στοιχείων και να βρίσκει ποιο από αυτά τα στοιχεία επιστρέφει true 4.2.

Εδώ θα δείξουμε το πιο απλό παράδειγμα του Αλγορίθμου: Έστω ότι έχουμε μια συνάρτηση:

$$f : \{0, 1\}^n \rightarrow \{0, 1\}, \text{ με } 2^n \text{ πιθανούς αριθμούς}$$

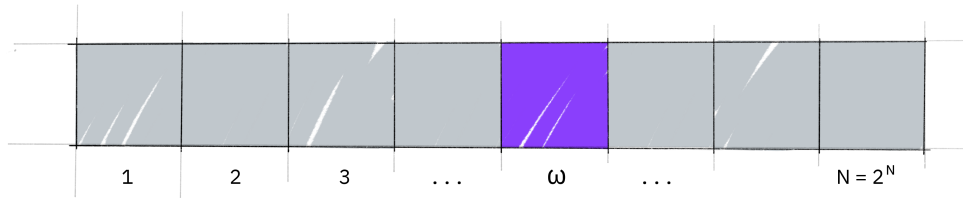


Figure 4.2: Grover's elements list



Figure 4.3: Grover's o cards deck

Ένα απλό παράδειγμα είναι για 2 qubits με ένα σύνολο από 4 αριθμούς 00, 01, 10, 11 και  $x_0 = 10$  το στοιχείο για το οποίο θέλουμε να επιστρέφει η συνάρτηση true όταν το συναντήσει.

$$f(x) \begin{cases} 1, & \text{if } x = 10 \\ 0, & \text{if } x \neq 10 \end{cases}$$

Παρατηρούμε πως χρειαζόμαστε αυτό που αποκαλούμε μια **Quantum Oracle** [22] δηλαδή μια συνάρτηση η οποία θα είναι Κβαντική, θα είναι δηλαδή αναστρέψιμη, αφού δεν μπορούμε να βάλουμε μια απλή συνάρτηση σαν είσοδο σε Κβαντικό Κύκλωμα και θα συνδυάζεται με την επιθυμητή για εμάς συνάρτηση. **Αυτό είναι ένα σημαντικό βήμα στη λογική του Κβαντικού Προγραμματισμού. Ο σχεδιασμός μιας τέτοιας Quantum Oracle.**

Η παρακάτω συνάρτηση πληρεί τις προϋποθέσεις:

$$O|x\rangle = (-1)^{f(x)}|x\rangle$$

Παρατηρούμε πως για αυτή την Quantum Oracle πετυχαίνουμε αντιστροφή του amplitude για την τιμή του στοιχείου που μας ενδιαφέρει. Αν την μοντελοποιήσουμε σαν Κβαντική Πύλη παίρνουμε:

$$O = \begin{pmatrix} |00\rangle & |01\rangle & |10\rangle & |11\rangle \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{matrix} |00\rangle \\ |01\rangle \\ |10\rangle \\ |11\rangle \end{matrix}$$

Αν εφαρμόσουμε αυτή την Κβαντική Πύλη στην αρχική μας είσοδο των τεσσάρων στοιχείων, έχουμε

$$\begin{aligned} |\psi_1\rangle &= H^{\otimes 2} |\psi_0\rangle = \frac{1}{\sqrt{2^2}}(|0\rangle + |1\rangle) \otimes (|0\rangle + |1\rangle) \\ &= \frac{1}{\sqrt{2^2}}(|00\rangle + |01\rangle + |10\rangle + |11\rangle) = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \end{bmatrix}^T \\ |\psi_2\rangle &= O |\psi_1\rangle = \frac{1}{\sqrt{2^2}}(|00\rangle + |01\rangle - |10\rangle + |11\rangle) = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} & -\frac{1}{2} & \frac{1}{2} \end{bmatrix}^T \end{aligned}$$

### Inversion Around the Mean

Τι έχουμε πετύχει ως τώρα; Όχι πολλά. Απλά αντιστρέψαμε το Amplitude για το στοιχείο που μας ενδιαφέρει, δηλαδή το στοιχείο 3. Αλλά αυτό δεν μπορούμε να το γνωρίζουμε αν κάνουμε τώρα τη μέτρηση, καθώς θα πάρουμε το απόλυτο των τιμών και άρα θα δούμε όλα τα Amplitudes ίδια.

Σε αυτό το σημείο εισάγουμε μια τεχνική που αποκαλείται **Inversion Around the middle** [23]. Το όνομα της τεχνικής μαρτυρά της λειτουργία της. Παίρνει ένα σύνολο αριθμών, βρίσκει το mean και τις αποστάσεις όλων των στοιχείων από αυτό το mean, και αντιστρέφει τις θέσεις των στοιχείων κρατώντας το mean ίδιο.

Για να το πετύχουμε στο συγκεκριμένο παράδειγμα με αυτά τα 4 στοιχεία, χρησιμοποιούμε τον κάτωθι πίνακα A.

$$A = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \end{bmatrix}$$

Και τον ακόλουθο πίνακα που κατασκευάζουμε με τη χρήση του Identity Matrix

$$2A - I^{\otimes 2} = \begin{bmatrix} -\frac{1}{2} & \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & -\frac{1}{2} & \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & \frac{1}{2} & -\frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & \frac{1}{2} & \frac{1}{2} & -\frac{1}{2} \end{bmatrix}$$

Τώρα εφαρμόζουμε τον νέο αυτό πίνακα που κατασκευάσαμε στην έξοδο του προηγούμενου βήματος.

$$|\psi_3\rangle = (2A - I^{\otimes 2}) |\psi_2\rangle = \begin{bmatrix} -\frac{1}{2} & \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & -\frac{1}{2} & \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & \frac{1}{2} & -\frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & \frac{1}{2} & \frac{1}{2} & -\frac{1}{2} \end{bmatrix} \begin{bmatrix} \frac{1}{2} \\ \frac{1}{2} \\ -\frac{1}{2} \\ \frac{1}{2} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}$$

Βουαλά! Για το συγκεκριμένο παράδειγμα των τεσσάρων στοιχείων με μόλις μία επανάληψη έχουμε σαν έξοδο το επιθυμητό αποτέλεσμα. Ο αλγόριθμός μας εντοπίζει στη θέση 3 το στοιχείο 10 το οποίο αναζητούσαμε.

Ας δούμε τώρα ένα παράδειγμα του ίδιου αλγορίθμου για ένα σύνολο οχτώ στοιχείων. Παραλήπουμε τα βήματα που φέρνουμε τα στοιχεία σε Superposition με χρήση Hadamard Gates και δείχνουμε μόνο τα βήματα που κάνουμε το Phase inversion και το Inversion around the mean

$$\begin{aligned} |\psi_2\rangle &= \left[ \frac{\sqrt{2}}{4} \quad -\frac{\sqrt{2}}{4} \quad \frac{\sqrt{2}}{4} \quad \frac{\sqrt{2}}{4} \quad \frac{\sqrt{2}}{4} \quad \frac{\sqrt{2}}{4} \quad \frac{\sqrt{2}}{4} \quad \frac{\sqrt{2}}{4} \right]^T \\ \Rightarrow |\psi_3\rangle &= \left[ \frac{\sqrt{2}}{8} \quad \frac{5\sqrt{2}}{8} \quad \frac{\sqrt{2}}{8} \quad \frac{\sqrt{2}}{8} \quad \frac{\sqrt{2}}{8} \quad \frac{\sqrt{2}}{8} \quad \frac{\sqrt{2}}{8} \quad \frac{\sqrt{2}}{8} \right]^T \\ \Rightarrow |\psi_4\rangle &= \left[ \frac{\sqrt{2}}{8} \quad -\frac{5\sqrt{2}}{8} \quad \frac{\sqrt{2}}{8} \quad \frac{\sqrt{2}}{8} \quad \frac{\sqrt{2}}{8} \quad \frac{\sqrt{2}}{8} \quad \frac{\sqrt{2}}{8} \quad \frac{\sqrt{2}}{8} \right]^T \\ \Rightarrow |\psi_5\rangle &= \left[ \frac{\sqrt{2}}{16} \quad -\frac{11\sqrt{2}}{16} \quad \frac{\sqrt{2}}{16} \quad \frac{\sqrt{2}}{16} \quad \frac{\sqrt{2}}{16} \quad \frac{\sqrt{2}}{16} \quad \frac{\sqrt{2}}{16} \quad \frac{\sqrt{2}}{16} \right]^T \end{aligned}$$

Παρατηρούμε πως στο τελευταίο βήμα το Amplitude του στοιχείου που θέλουμε να εντοπίσουμε (στην προκειμένη περίπτωση το στοιχείο 001), δηλαδή το δεύτερο είναι αισθητά μεγαλύτερο από τα υπόλοιπα, έτσι σε μια μέτρηση είναι πιθανότερο το σύστημα να καταρρεύσει στη αυτή τη θέση.

Αν τώρα επαναλάβουμε την εκτέλεση αρκετές φορές τότε τις περισσότερες το σύστημα θα καταρρέει στην επιθυμητή θέση. Γενικά στον Αλγόριθμο του Grover συνίσταται να επαναλαμβάνουμε την εκτέλεση  $\pi/4\sqrt{2^n}$  φορές για να οδηγηθούμε στο επιθυμητό αποτέλεσμα.

## Συμπέρασμα

Μπορούμε τώρα να πούμε με σιγουριά πως έχουμε κατανοήσει τη λογική σχεδιασμού ενός Κβαντικού Αλγορίθμου και του κατάλληλου κυκλώματος αλλά και τα πλεονεκτήματά του. Μπορούμε να εφαρμόζουμε αλγόριθμους που λειτουργούν σε τόσα στοιχεία όσα μας επιτρέπουν τα qubits που έχουμε, ήτοι  $2^n$ . Έστερα μπορούμε να εφαρμόζουμε συναρτήσεις που καλούμε Quantum Oracles **ταυτόχρονα και παράλληλα** σε όλα τα στοιχεία προκειμένου να βγάλουμε τα συμπεράσματά μας. Μοντελοποιούμε αυτές τις Quantum Oracles σαν Κβαντικές Πύλες, αντιστρέψιμους δηλαδή πίνακες.

Η όποια πολυπλοκότητα προκύπτει κάθε φορά από το πόσες φορές θα πρέπει να επαναλάβουμε κάποια βήματα της εκτέλεσης ή ολόκληρη την εκτέλεση ή τη μέτρηση ή και συνδιασμό των παραπάνω προκειμένου να συγκλίνουμε στο επιθυμητο αποτέλεσμα. Θα δούμε στην πορεία πως μπορούμε χρησιμοποιώντας τα διαθέσιμα qubits να έχουμε αρχικές καταστάσεις που αντιπροσωπεύουν αντικειμενικές μεταβλητές ενός πραγματικού προβλήματος και όχι απλά στοιχεία τύπου 00, 01, 10, 11. Αυτό επιτυγχάνεται με χρήση phase shifting λογικών πυλών.

Στη συνέχεια θα προχωρήσουμε στην παρουσίαση δύο κβαντικών αλγορίθμων των οποίων εξετάσαμε και πειραματικά. Του Quantum Fourier Transformation και του Quantum Kmeans. Ο τελευταίος αποτελεί μια υβριδική δική μας πρόταση.

## 4.2 Quantum Fourier Transformation

Ο Quantum Fourier Transformation [24] είναι το Κβαντικό ανάλογο του Κλασσικού Discrete Fourier Transformation [25]. Στον DFT παίρνουμε ένα σύνολο στοιχείων στο πεδίο του χρόνου και βρίσκουμε αντίστοιχο αριθμό άλλων στοιχείων στο πεδίο της συχνότητας ή αντίστροφα, χρησιμοποιώντας μετασχηματισμούς.

$$x = (x_1, x_2, \dots, x_n) \text{ to } y = (y_1, y_2, \dots, y_n)$$

Για μετασχηματισμό από το πεδίο του χρόνου στο πεδίο της συχνότητας

$$\psi_k = \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} x_j e^{2\pi i \frac{kj}{N}}$$

Το Κβαντικό ανάλογο του DFT παίρνει σαν είσοδο μια Quantum State με κάποια Amplitudes και παράγει σαν έξοδο μια άλλη Quantum State με διαφορετικά Amplitudes. Χρησιμοποιώντας την παραπάνω λογική ως θεωρήσουμε τον μετασχηματισμό των Amplitudes από το  $a_j$  στο  $b_k$ :

$$b_k = \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} a_j e^{2\pi i \frac{kj}{N}}$$

Έτσι αρκεί απλά πρέπει να φτιάξουμε τον παρακάτω Κβαντικό μετασχηματισμό:

$$|j\rangle = \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} a_j e^{2\pi i \frac{kj}{N}} |k\rangle$$

Για να το πετύχουμε αυτό πρέπει να φτιάξουμε τη γνωστή μας πλέον Quantum Oracle που θα εφαρμοστεί στην είσοδό μας. Σύμφωνα με τη βιβλιογραφία η παρακάτω συνάρτηση ικανοποιεί το επιθυμητό αποτέλεσμα.

$$F = \frac{1}{\sqrt{N}} \begin{bmatrix} 1 & 1 & 1 & \dots & 1 \\ 1 & \omega & \omega^2 & \dots & \omega^{N-1} \\ 1 & \omega^2 & \omega^4 & \dots & \omega^{2(N-1)} \\ \vdots & \vdots & \vdots & \dots & \vdots \\ 1 & \omega^{N-1} & \omega^{2(N-1)} & \dots & \omega^{(N-1)(N-1)} \end{bmatrix} \text{ με } \omega = e^{(2\pi i/n)}$$

Μπορεί να μοιάζει λίγο τρομακτική στην όψη αλλά είναι αρκετά απλή όπως θα δούμε στα ακόλουθα παραδείγματα.

Ας πάρουμε αρχικά ένα σύστημα τεσσάρων qubit:

$$|\psi\rangle = a_0 |0\rangle + a_1 |1\rangle + a_2 |2\rangle + a_3 |3\rangle = a_0 |00\rangle + a_1 |01\rangle + a_2 |10\rangle + a_3 |11\rangle$$

Χρησιμοποιώντας τον μετασχηματισμό θα έχουμε:

$$F |\psi\rangle = |\phi\rangle = \sum_{k=0}^3 b_k |k\rangle$$

Όπου υπολογίζουμε τα  $b_k$  ακολουθώντας:

$$\begin{aligned} b_0 &= \frac{1}{2}(a_0 + a_1 + a_2 + a_3) \\ b_1 &= \frac{1}{2}(a_0 + a_1 e^{i\frac{\pi}{2}} + a_2 e^{i\pi} + a_3 e^{i\frac{3\pi}{2}}) \\ b_2 &= \frac{1}{2}(a_0 + a_1 e^{i\pi} + a_2 e^{2i\pi} + a_3 e^{3i\pi}) \\ b_3 &= \frac{1}{2}(a_0 + a_1 e^{3i\frac{\pi}{2}} + a_2 e^{3i\pi} + a_3 e^{2i\frac{\pi}{2}}) \end{aligned}$$

Εύκολα κατασκευάζουμε την Quantum Oracle ακολουθώντας:

$$F = \frac{1}{2} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & \omega & \omega^2 & \omega^3 \\ 1 & \omega^2 & \omega^4 & \omega^6 \\ 1 & \omega^3 & \omega^6 & \omega^9 \end{bmatrix} = \frac{1}{2} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & i & -1 & -i \\ 1 & -1 & 1 & -i \\ 1 & -i & -1 & i \end{bmatrix}$$

Και αν την εφαρμόσουμε στην είσοδό μας παίρνουμε:

$$F |\psi\rangle = \frac{1}{2} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & i & -1 & -i \\ 1 & -1 & 1 & -i \\ 1 & -i & -1 & i \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{bmatrix} = \frac{1}{2} \begin{bmatrix} a_0 + a_1 + a_2 + a_3 \\ a_0 + ia_1 - a_2 - ia_3 \\ a_0 - a_1 + a_2 - a_3 \\ a_0 - ia_1 - a_2 + ia_3 \end{bmatrix}$$

Συμπεράσματα

Αρχικά παρατηρούμε πως μοντελοποιούμε  $2^n$  κλασσικά bits με  $n$  qubits. επίσης πως στον QFT η χρονική πολυπλοκότητα δεν καθορίζεται από το πόσες επαναλήψεις θα εκτελέσουμε καθώς με μία μόνο επανάληψη φτάνουμε στο επιθυμητό αποτέλεσμα. Εδώ η χρονική πολυπλοκότητα προκύπτει βέβαια από τον αριθμό των λογικών πυλών που πρέπει να χρησιμοποιήσουμε για να τελεστούν όλοι οι μετασχηματισμοί.

Συγκεκριμένα θέλουμε  $n(n-1)/2$  πύλες που σημαίνει  $O(n^2)$  σε χρονική πολυπλοκότητα. Τεράστια βελτίωση με τον κλασσικό DFT που έχει  $O(n^{2^n})$  χρονική πολυπλοκότητα.

Αυτά βέβαια σε θεωρητικό επίπεδο. Θα δούμε μετά στην εκτέλεση σε πραγματικό Κβαντικό Υπολογιστή πως τα έχουμε ακόμα αρκετά βήματα να κάνουμε μέχρι να φτάσουμε τις επιθυμητές δυνατότητες σε επίπεδο εκτέλεσης.

### 4.3 A Quantum kmeans approach

Ο kmeans [26] είναι ένας αλγόριθμος Μηχανικής Μάθησης [27] με σκοπό την ομαδοποίηση (clustering) ενός συνόλου δεδομένων σε  $k$  υποομάδες. Τα υποσύνολα διαχωρίζονται βάσει του mean της απόστασής τους από  $k$  centroids, 4.4.

Ο αλγόριθμος του kmeans είναι αρκετά απλός στο σχεδιασμό του όπως βλέπουμε 1:

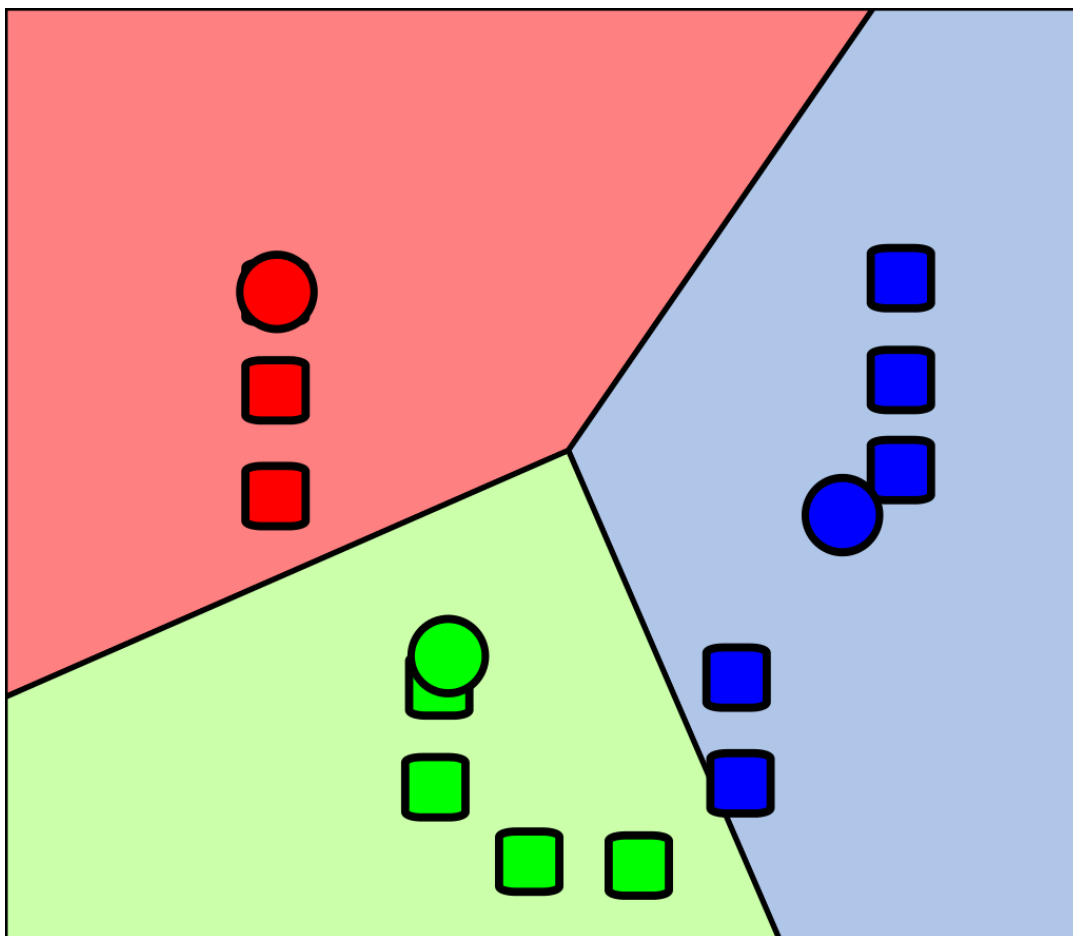


Figure 4.4: A simple kmeans clustering example

---

**Algorithm 1** kmeans Algorithm
 

---

- 1: Determine number of centroids ( $K$ )
  - 2: Randomly select  $K$  distinct centroids
  - 3: Measure the distance (euclidean distance) between each point and the centroid
  - 4: Assign the each point to the nearest cluster
  - 5: Calculate the mean of each cluster as new centroid
  - 6: Repeat step 3–5 with the new center of cluster
  - 7: Calculate the variance of each cluster
  - 8: Repeat step 2–7 until get the lowest sum of variance
-



### 4.3.1 Swap Test

Το πρώτο βήμα για την παρουσίαση του Κβαντικού kmeans είναι να παρουσιάσουμε την ρουτίνα Swap Test [28] μια πολύ γνωστή ρουτίνα στον Κβαντικό Προγραμματισμό που σου προσφέρει μια εκτίμηση για την "ομοιότητα" δύο states, δηλαδή το overlap μεταξύ δύο states  $\langle a|b \rangle$ . Ο αλγόριθμος πάει ως εξής 2:

---

#### Algorithm 2 Swap test Algorithm

---

- 1: Initialize two states  $|a\rangle, |b\rangle$  of  $n$  qubits each and a control qubit  $|0\rangle$ :  
 $|\psi_0\rangle = |0, a, b\rangle$
- 2: Apply Handamard gate on control qubit:  
 $|\psi_1\rangle = H \otimes I^{\otimes n} I^{\otimes n} |\psi_0\rangle = \frac{1}{\sqrt{2}}(|0, a, b\rangle + |1, a, b\rangle)$
- 3: Apply controlled SWAP gate on  $|a\rangle, |b\rangle$  that swaps  $a, b$  if the controll qubit is 1:  
 $|\psi_2\rangle = \frac{1}{\sqrt{2}}(|0, a, b\rangle + |1, b, a\rangle)$
- 4: Apply second Handamard gate on controll qubit:  
 $|\psi_3\rangle = \frac{1}{2}|0\rangle(|a, b\rangle + |b, a\rangle) + \frac{1}{2}|1\rangle(|a, b\rangle - |b, a\rangle)$
- 5: Perfom Measurment

$$\begin{aligned}
 P(|0\rangle) &= |\frac{1}{2}\langle 0|0\rangle(|a, b\rangle + |b, a\rangle) + \frac{1}{2}\langle 0|1\rangle(|a, b\rangle - |b, a\rangle)|^2 \\
 &= \frac{1}{4}(|a, b\rangle + |b, a\rangle)^2 \\
 &= (\langle b|b\rangle \langle a|a\rangle + \langle b|a\rangle \langle a|b\rangle + \langle a|b\rangle \langle b|a\rangle + \langle a|a\rangle \langle b|b\rangle) \\
 &= \frac{1}{2} + \frac{1}{2}|\langle a|b\rangle|^2
 \end{aligned}$$


---

Τι παρατηρούμε από τον αλγόριθμο 2; Ότι στο τέλος παίρνουμε σαν αποτέλεσμα έναν πραγματικό αριθμό που προκύπτει από το εσωτερικό γινόμενο των  $|a\rangle, |b\rangle$ . Άρα αν η μέτρηση μας επιστρέψει 0.5 τότε οι δύο καταστάσεις έχουν μηδενικό εσωτερικό γινόμενο, άρα είναι ορθόγones, ενώ αν μας επιστρέψει 1 τότε οι δύο καταστάσεις είναι πανομοιότυπες. Προφανώς το μεσοδιάστημα μεταξύ 0.5 – 1 μας προσφέρει ένα σαφές κριτήριο για την ομοιότητα των δύο καταστάσεων.

Αυτή την πολύ σημαντική ιδιότητα θα χρησιμοποιήσουμε για να βρούμε το αντίστοιχο της Ευκλείδειας Απόστασης [29] στο Κβαντικό πεδίο. Αν μπορούμε να βρούμε την ομοιότητα δύο καταστάσεων, τότε θα πρέπει να μοντελοποιήσουμε δύο σημεία στο Καρτεσιανό σύστημα με τέτοιο τρόπο στο Κβαντικό πεδίο ώστε, η ομοιότητα των καταστάσεων τους που θα δημιουργήσουμε να μαρτυρά πως έχουν μικρότερη απόσταση στο αρχικό Καρτεσιανό σύστημα.

### 4.3.2 Κβαντική αναπαράσταση ενός Κλασσικού state

Σε αυτό το σημείο θα πρέπει να βρούμε μια αναπαράσταση στο Κβαντικό Μοντέλο, για την είσοδο των Καρτεσιανών σημείων μας. Έστω ότι έχουμε δύο σημεία

$$x = [x_1 \ x_2]^T, \psi = [\psi_1 \ \psi_2]^T$$

Προτείνουμε και εφαρμόζουμε τον εξής μετασχηματισμό που μεταφέρει τα δύο αυτά σημεία πάνω στην Bloch Sphere 2.2, διατηρώντας τις ιδιότητες των μεταξύ τους αποστάσεων ως προς

το μέτρο της ομοιότητας καταστάσεων. Με λίγα λόγια αν αυτά τα δύο σημεία απέχουν περισσότερο στο Καρτεσιανό σύστημα τότε παρουσιάζουν μεγαλύτερη ανομοιότητα στην Κβαντική Αναπαράσταση:

$$\begin{aligned}\theta_1 &= (x_0 + 1)\frac{p_i}{2}, \phi_1 = (x_1 + 1)\frac{p_i}{2} \\ \theta_2 &= (\psi_0 + 1)\frac{p_i}{2}, \phi_2 = (\psi_1 + 1)\frac{p_i}{2} \\ \Rightarrow |x\rangle &= [\theta_1 \quad \phi_1], |\psi\rangle = [\theta_2 \quad \phi_2]\end{aligned}$$

Το μόνο που μένει είναι να φέρουμε στο σύστημα στην επιθυμητή Κβαντική κατάσταση, το οποίο θα επιτύχουμε με Quantum Rotation Gates όπως θα δούμε στην υλοποίηση στο επόμενο κεφάλαιο. Το ερώτημα είναι αν έχουμε datasets με πολλές διαστάσεις και όχι μόνο δύο, τότε πώς μπορούμε να κάνουμε την Κβαντική αναπαράσταση των Classical bits; Σε αυτό το ερώτημα απαντάει αυτή η πολύ ενδιαφέρουσα δουλειά [30] όπου επιτυγχάνει Κβαντική αναπαράσταση για είσοδο με πολλές διαστάσεις.

Τέλος να αναφέρουμε πως η σημαντική ρουτίνα για τη χρονική πολυπλοκότητα είναι η ρουτίνα που υπολογίζει την ομοιότητα ή απόσταση δύο σημείων όπου στην υλοποίησή μας είναι  $O(\log N)$  (γιατί χρειαζόμαστε  $\log N$  Qubits για  $n$  χαρακτηριστικά εισόδου) ενώ στο Κλασσικό ανάλογο η πολυπλοκότητα είναι της τάξεως του  $n$ .

# Qiskit, IBM Quantum Experience - Υλοποίηση σε πραγματικό Κβαντικό Υπολογιστή και πειράματα

*If you are not completely confused by quantum mechanics, you do not understand it*

*John Wheeler*

## 5.1 Qiskit

Το Qiskit [31] είναι ένα εργαλείο SDK για να δουλεύουμε σε Κβαντικό Hardware σε επίπεδο αλγορίθμων, κβαντικών πυλών και κυκλωμάτων. Χρησιμοποιώντας το μπορούμε μέσω απλών API όπως της Python να σχεδιάζουμε Κβαντικά κυκλώματα, τα οποία υλοποιούν Κβαντικούς αλγορίθμους για να εξάγουμε τα επιθυμητά αποτελέσματα και να τα τεστάρουμε πάνω σε προσομοιωτές είτε τοπικά είτε σε ειδικά σχεδιασμένα clusters. Μπορούμε επίσης να αιτηθούμε εκτέλεση σε πραγματικό Κβαντικό Hardware το οποίο μας δίνει η IBM [32] μέσω του IBM Quantum Experience. Κατά την εκτέλεση των πειραμάτων αυτής της εργασίας δεν δυνόμεθα να πάρουμε Κβαντικό Hardware με πάνω από 5 qubits. Τα οποία όμως ήταν και υπερασκετά για τη δική μας υλοποίηση.

Το κύριο API του Qiskit είναι σε Python αλλά έχουν πλέον βγει εκδοχές και σε Javascript και σε Swift. Υπάρχουν διαθέσιμα στο cluster της IBM Jupyter notebooks με παραδείγματα και υλοποιήσεις για να μπορέσει κάποια γρήγορα να αποκτήσει οικιότητα με τον προγραμματισμό σε Κβαντικό Υπολογιστή.

### Qiskit Components

Ένας από τους βασικούς components του Qiskit είναι το **Qiskit Terra** [33]. Είναι το εργαλείο πάνω στο οποίο είναι χτισμένο όλο το λογισμικό του Qiskit. Παρέχει τα κατάλληλα εργαλεία για να φτιάχνουμε Κβαντικά κυκλώματα χρησιμοποιώντας qubits και Κβαντικές πύλες. Είναι αυτό που μας επιτρέπει να γράφουμε κώδικα που χρησιμοποιεί αποκλειστικά Κβαντικές πύλες και είναι πολύ κοντά σε Κβαντική Γλώσσα Μηχανής.

```
: import matplotlib.pyplot as plt
%matplotlib inline
import numpy as np
from math import pi
from qiskit.tools.visualization import circuit_drawer
```

Figure 5.1: Qiskit Code 1

```
# Import Aer
from qiskit import Aer
from qiskit import IBMQ
```

Figure 5.2: Qiskit Code 2

Σημαντικό component του Qiskit είναι το Qiskit Aer [34] το οποίο μας παρέχει Κβαντικούς προσομοιωτές που τρέχουν πάνω σε HPC resources-Clusters προκειμένου να προγραμματίζουμε και να τεστάρουμε σε Κβαντικό επίπεδο. Μας παρέχει επίσης εργαλεία που μπορούμε να προσομοιώνουμε και κοντά στον πραγματικό Κβαντικό θόρυβο, αφού όπως έχουμε τονίσει τα πραγματικά Κβαντικά συστήματα είναι ακόμα αρκετά ασταθή και εμπεριέχουν πολύ θόρυβο.

Το IBM Quantum Provider [35] μας δίνει τη δυνατότητα αντίστοιχα με το Qiskit Aer να επιλέξουμε μέσα από μια πληθώρα Clusters με πραγματικό Κβαντικό Hardware για να τρέξουμε τον κώδικά μας. Μας δίνει τη δυνατότητα να φιλτράρουμε και να επιλέξουμε την κατάλληλη για εμάς πλατφόρμα

## Εισαγωγή στο Qiskit

Θα δούμε τώρα πως είναι η εμπειρία του να προγραμματίζει σε Κβαντικό επίπεδο χρησιμοποιώντας τα Jupyter notebooks που μας παρέχει το Qiskit. Στο παράδειγμα που βλέπουμε στις εικόνες 5.1, 5.2, 5.3, παρατηρούμε την ευκολία που μας δίνει το Qiskit για να προγραμματίζουμε σε Κβαντικό επίπεδο, σε Jupyter notebooks με τα οποία έχουμε οικιότητα. Αρχικά κάνουμε import τα μοντέλα που θα χρειαστούμε, ανάμεσά τους και τους components του Qiskit τους οποίους αναφέραμε παραπάνω. Ύστερα, επειδή προγραμματίζουμε σε local επίπεδο, πρέπει την πρώτη φορά να αποθηκεύσουμε το apikey που μας δίνει το IBM Quantum Services το οποίο σε κάθε επόμενη εκτέλεση κάνουμε load από τον τοπικό δίσκο για να έχουμε πρόσβαση στα clusters της IBM.

Στο συγκεκριμένο παράδειγμα δείχνουμε πως μπορούμε να δούμε μια λίστα από τους διαθέσιμους actual quantum providers και να φιλτράρουμε τα συγκεκριμένα χαρακτηριστικά που θέλουμε (πχ πανω από 5 qubits και σε λειτουργία αυτή τη στιγμή) και ύστερα να χρησιμοποιήσουμε το backend αυτού του provider που θέλουμε. Σημειώνεται στα σχόλια είναι η αντίστοιχη διαδικασία που θα χρησιμοποιούσαμε αν επιλέγαμε να δουλέψουμε με προσομοιωτή. Είθισται το μεγαλύτερο μέρος του προγραμματισμού σε Κβαντικό επίπεδο να γίνεται σε προσομοιωτή.

## 5.2 Quantum Discrete Fourier Transformation on Qiskit

Σε αυτό το σημείο θα προχωρήσουμε προς την πρώτη μας υλοποίηση, του QFT που είδαμε στο 4.2, για 4 qubits. Βλέπουμε ένα κομμάτι της κατασκευής του κυκλώματος στην εικόνα 5.4 και το κύκλωμα που δημιουργείται στην εικόνα 5.5.

Στον κώδικα βλέπουμε πως αρχικά παίρνουμε το πρώτο qubit από μια Hadamard gate

```

IBMQ.save_account(apikey, overwrite=True)

IBMQ.load_account() # Load account from disk
IBMQ.providers()    # List all available providers

ibmqfactory.load_account:WARNING:2022-02-11 15:21:31,806: Credentials are already in use. The existing account in
the session will be replaced.

[<AccountProvider for IBMQ(hub='ibm-q', group='open', project='main')>]

provider=IBMQ.get_provider(hub='ibm-q')

for i in provider.backends(filters=lambda x: int(x.configuration().n_qubits) >= 5
                             and not x.configuration().simulator
                             and x.status().operational==True):
    print(i)

ibmq_bogota
ibmq_lima
ibmq_belem
ibmq_quito
ibmq_manila

# To use local qasm simulator or an actual Quantum backend
# simulator
#backend = Aer.get_backend('qasm_simulator')

#actual quantum

backend = provider.get_backend('ibmq_bogota')

```

Figure 5.3: Qiskit Code 3

ύστερα από `cui` gates μαζί με τα άλλα qubits η οποία είναι μια phase shifting gate και φέρνει τα qubits σε τέτοια θέση μεταξύ τους ώστε να έχουν σχετική γωνία ίση με την τιμή που δίνουμε. Έπεται βάζουμε ένα `qc.barrier()` που φροντίζει να μην προχωρήσει η εκτέλεση εάν δεν έχουν ολοκληρωθεί όλες οι διαδικασίες έως εκείνο το σημείο. Ο υπόλοιπο κώδικας συνεχίζει αναλόγως, ώστε στο τέλος να έχουμε το επιθυμητό μετασχηματισμένο state.

### Πειραματική αξιολόγηση

Τρέχουμε τον παραπάνω κωδικά για να βρούμε τον μετασχηματισμό σε ένα διάνυσμα αρχικοποιημένο με 4 μηδενικά. Ο μετασχηματισμός που παίρνουμε σαν αποτέλεσμα είναι αυτός στην εικόνα 5.6. Τρέχουμε για την ίδια είσοδο τον κλασσικό DFT 5.7, σε μια GPU (Nvidia Geforce 1650 Ti)

Το αποτέλεσμα που λαμβάνουμε από την Κβαντική εκτέλεση είναι σωστό. Είναι σημαντικό να γνωρίζουμε πως η Κβαντική μας υλοποίηση προσφέρει σωστά αποτελέσματα.

Αν θέλαμε να βρούμε το μετασχηματισμό για κάθε qubit αρκεί να μετρήσουμε το κάθε qubit αρκετές φορές μέχρι ο μέσος όρος να σταθεροποιηθεί γύρω από μια τιμή που θα είναι και η σωστή τιμή του μετασχηματισμού. Όμως.

Χρησιμοποιούμε τη συνάρτηση `time_taken()` που μας δίνει το Qiskit για να υπολογίσουμε το χρόνο εκτέλεσης. Αντίστοιχα υπολογίζουμε και το χρόνο εκτέλεσης της κλασσικής μεθόδου. Παρότι στη θεωρητική ανάλυση που κάναμε στο 4.2 είδαμε πως ο QFT έχει χρονική πολυπλοκότητα  $O(n^2)$  ενώ ο DFT  $O(n2^n)$  αυτό δεν αποτυπώνεται στα αποτελέσματά μας. Στην πράξη βλέπουμε πως ο χρόνος εκτέλεσης στον Κβαντικό provider είναι 3 – 4s ενώ ο αντίστοιχος DFT παίρνει  $2^{-5e}$ .

Παρατηρούμε μια τεράστια διαφορά στη χρονική εκτέλεση. Αρχικά να πούμε πως η πολύ μικρή είσοδός μας δεν μας επιτρέπει να δούμε τις επιπτώσεις του μεγέθους  $n$  στην εκτέλεση, αλλά όπως προείπαμε δεν έχουμε προσβαση με πολλά qubits. Στο τέλος του κεφαλαίου θα αναλύσουμε τα συμπεράσματά μας για την αισθητά αργή εκτέλεση στον Κβαντικό provider.

Πριν προχωρήσουμε σε συμπεράσματα όμως θα δούμε την υλοποίηση του δικού μας

```
qc = QuantumCircuit(q)
#-----
qc.h(0)
qc.cu1(pi/2,q[1],q[0])
qc.cu1(pi/4,q[2],q[0])
qc.cu1(pi/8,q[3],q[0])
#-----
qc.barrier()
qc.h(1)
qc.cu1(pi/2,q[2],q[1])
qc.cu1(pi/4,q[3],q[1])
#-----
qc.barrier()
qc.h(2)
qc.cu1(pi/2,q[3],q[2])
#-----
qc.barrier()
qc.h(3)
#-----|
circuit_drawer(qc, output='mpl', style={'backgroundcolor': '#EEEEEE'})
```

Figure 5.4: Qiskit QFT code

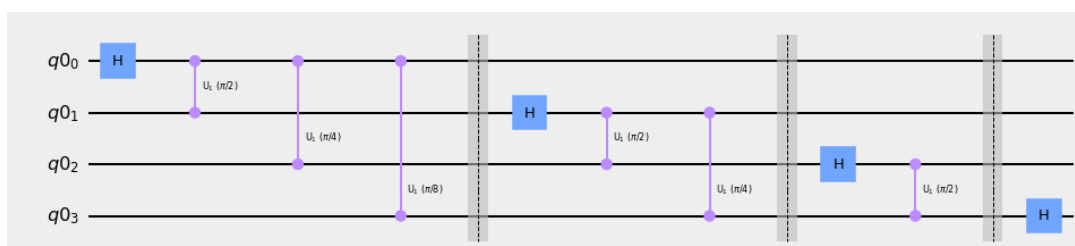


Figure 5.5: Qiskit QFT circuit for 4 Qubits

```
Statevector([0.25+0.j, 0.25+0.j, 0.25+0.j, 0.25+0.j, 0.25+0.j, 0.25+0.j,
            0.25+0.j, 0.25+0.j, 0.25+0.j, 0.25+0.j, 0.25+0.j, 0.25+0.j,
            0.25+0.j, 0.25+0.j, 0.25+0.j, 0.25+0.j],
            dims=(2, 2, 2, 2))
```

Figure 5.6: Result of QFT

```
import time
import numpy as np

import scipy as sp

def dftmtx(N):
    return sp.fft(sp.eye(N))

#classical
N=4
x = np.zeros(4, dtype=int) # x is any input data with those dimensions
W = dftmtx = np.fft.fft(np.eye(4))
print(x)
print(W)

timet=0.0
for i in range(1000):
    time_1=time.time()

    dft_of_x = W.dot(x).dot(W)

    time_2=time.time()
    timet=(time_2-time_1)+timet
```

Figure 5.7: Code of clasical DFT

kmeans όπου θα φανεί καλύτερα η επίδραση της εισόδου  $n$  .

### 5.3 A Quantum kmeans on Qiskit

Σε αυτό το σημείο θα δείξουμε την υλοποίηση του Κβαντικού kmeans που παρουσιάσαμε στο κεφάλαιο 4.3. Στην εικόνα 5.8 βλέπουμε το κομμάτι του κώδικα που υπολογίζει το κατά πόσο απέχουν δύο σημεία. Στην εικόνα 5.9 βλέπουμε το κύκλωμα που δημιουργείται. Στον κώδικα βλέπουμε πως αρχικά υπολογίζουμε τις γωνίες των μετασχηματισμών των κλασσικών σημείων πάνω στην Bloch Sphere. Ύστερα με τη χρήση της phase shifting gate  $U$  φέρνουμε τα δύο qubits που έχουμε καταχωρήσει σε κβαντικούς καταχωρητές στην κατάλληλη θέση πάνω στην Bloch Sphere. Ύστερα εφαρμόζουμε την swap test και ύστερα μετράμε τα δύο qubits για να βρούμε το μέτρο της απόστασης ή καλύτερα της ομοιότητας των μεταξύ τους καταστάσεων.

Εκτελούμε αυτό το κομμάτι κώδικα όσες φορές δίνουμε στην παράμετρο shots. Αυτό γίνεται γιατί όπως είπαμε η Κβαντική μέτρηση είναι πιθανοτική, οπότε πρέπει να εκτελέσουμε αρκετές φορές μια ρουτίνα και να πάρουμε το μέσο όρο των αποτελεσμάτων που θα συγκρίνει στη σωστή τιμή. Βλέπουμε τα αποτελέσματα του μέτρου της ομοιότητας δύο states που μας δίνει η swap test για διαφορετικό αριθμό shots στα 5.10, 5.10a, 5.10b, 5.10c.

Τώρα ας εξετάσουμε την ορθότητα του αλγορίθμου μας για ομαδοποίηση διαφορετικού

```
def newN_Distance(x,y, Nshots=1024):
    phi_x = ((x[0] + 1) * pi / 2)
    phi_y = ((y[0] + 1) * pi / 2)

    theta_x = ((x[1] + 1) * pi / 2)
    theta_y = ((y[1] + 1) * pi / 2)

    qreg = QuantumRegister(3, 'qreg')

    # ClassicalRegister to hold the result
    creg = ClassicalRegister(3, 'creg')

    qc = QuantumCircuit(qreg, creg, name='Quantum kmeans')

    qc.h(qreg[2])

    # Transform the initial state
    qc.u(theta_x, phi_x, 0, qreg[0])
    qc.u(theta_y, phi_y, 0, qreg[1])

    qc.cswap(qreg[2], qreg[0], qreg[1])

    # Apply second Hadamard to ancillary
    qc.h(qreg[2])

    # Measure ancillary
    qc.measure(qreg[2], creg[0])

    qc.reset(qreg)

    # Register and execute job
    job = execute(qc, backend=backend, shots=Nshots)
    result = job.result()

    data = result.data()

    time=result.time_taken

    return [data, time]
```

Figure 5.8: Portion of Code of Quantum kmeans

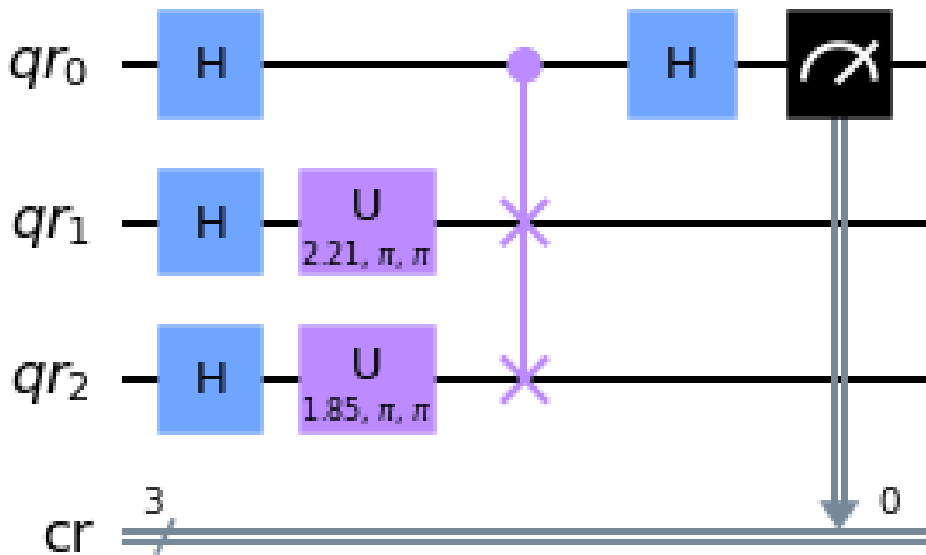


Figure 5.9: Quantum distance of 2 qubits circuit



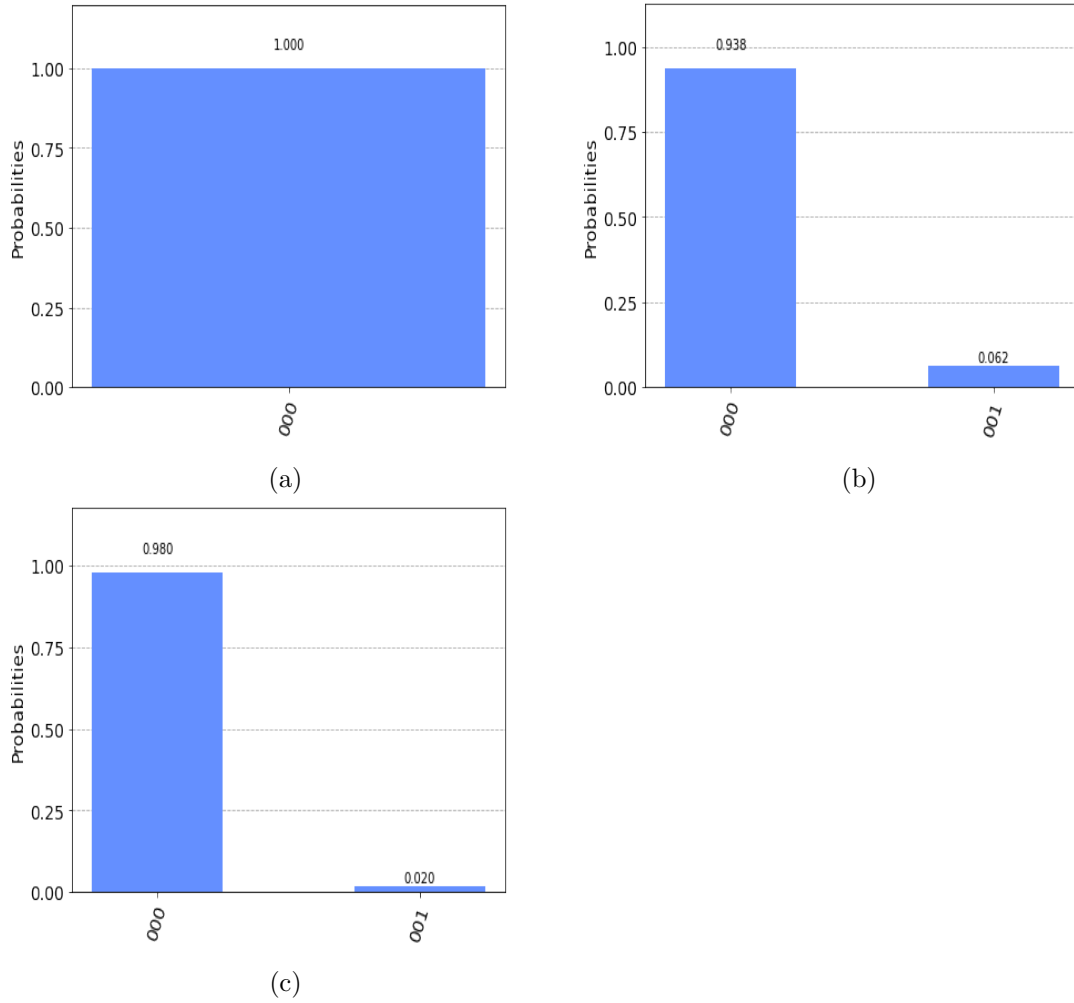


Figure 5.10: Ομοιότητα 2 states αναλόγως του αριθμού των shots

αριθμού εισόδου διασδιάστων σημείων, για 7 cendroids. Βλέπουμε τα αποτελέσματα προϊόντος του χρόνου στις εικόνες 5.11. 5.11a, 5.11b, 5.11c και αντίστοιχα για είσοδο 1000 σημείων στις εικόνες 5.12, 5.12a, 5.12b, 5.12c. Παρατηρούμε πως ο αλγόριθμος και η Κβαντική αναπαράσταση των Κλασσικών states των σημείων που επιλέξαμε, λειτουργεί ορθά και παράγει τα επιθυμητά αποτελέσματα.

### Πειραματική αξιολόγηση της απόδοσης του Αλγορίθμου μας

Για να μετρήσουμε το χρόνο εκτέλεσης με διαφορετικές παραμέτρους του πειράματος χρησιμοποιούμε τη συνάρτηση το Qiskit `time_taken()`. Χρησιμοποιούμε τον backend IBM `boqota` ενώ για τον κλασσικό kmeans μια GPU `Nvidia Geforce 1650 Ti`). Σαν χαρακτηριστικά που μας ενδιαφέρουν εξετάζουμε τον αριθμό των shots αλλά και των input features. Εδώ "κλέβουμε" όσον αφορά τα αποτελέσματα, όχι της χρονικής πολυπλοκότητας όμως. Όπως είπαμε αν έχουμε πχ είσοδος με 10 χαρακτηριστικά θα πρέπει να βρούμε μια Κβαντική αναπαράσταση για είσοδο με 10 χαρακτηριστικά, μια διαδικασία αρκετά απαιτητική που ξεπερνά τις ανάγκες αυτής τη εργασίας. Χωρίς να χάνεται κάτι σε εγκυρότητα, δίνουμε σε σημεία τυχαία χαρακτηριστικά και υπολογίζουμε το μέτρο της μεταξύ τους ομοιότητας, για να δούμε και το κατά πόσο επηρεάζεται ο χρόνος εκτέλεσης στο Κβαντικό υλικό, από το μέγεθος της

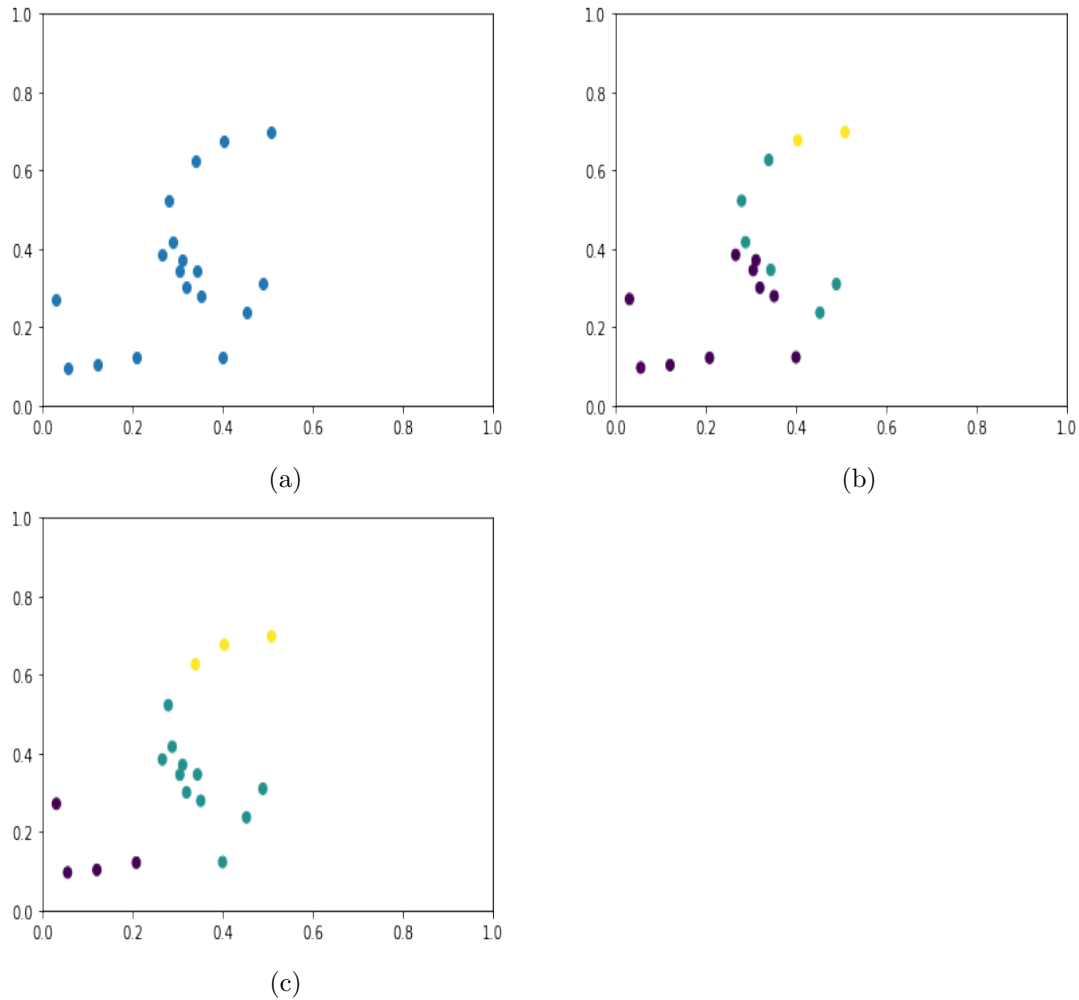


Figure 5.11: Ομαδοποίηση 20 σημείων προϊόντος του χρόνου

διάστασης των χαρακτηριστικών της εισόδου.

Στο τέλος καταλήγουμε με έναν dataset 5.13 που αποτελείται από τα Qtime, Ctime δηλαδή τους χρόνους εκτέλεσης στο Κβαντικό και το Κλασσικό μοντέλο αντίστοιχα, όπως και τα features, shots που είναι τα χαρακτηριστικά της εισόδου του kmeans και ο αριθμός των εκτελέσεων που κάνουμε στο Κβαντικό υλικό μέχρι να έχουμε ασφαλή αποτελέσματα.

Γνωρίζουμε διαισθητικά και από τα προηγούμενα πειράματα πως ο χρόνος εκτέλεσης στο Κβαντικό υλικό θα είναι πολύ μεγαλύτερος από τον αντίστοιχο στο Κλασσικό. Οπότε θέλουμε να κάνουμε μια ανάλυση ώστε να δούμε αν παρόλο το overhead που προκύπτει από την έλλειψη σταθερότητας ακόμα των Κβαντικών συστημάτων, υπάρχει κάποια συνάφεια της εισόδου με το χρόνο εκτέλεσης. Αρχικά βρίσκουμε ένα correlation matrix [36] με τη μέθοδο pearson 5.14. Παρατηρούμε μια μέτρια προς ασθενή συσχέτιση του μεγέθους της διάστασης των χαρακτηριστικών της εισόδου και του αριθμού εκτελέσεων στο Κβαντικό υλικό με το χρόνο εκτέλεσης. Έστερα κάνουμε ανάλυση με τη μέθοδο spearman 5.15 η οποία μας επιτρέπει να βλέπουμε συσχετίσεις μεταξύ δύο μεγεθών που δεν είναι μόνο γραμμικές. Εδώ βλέπουμε ισχυρή συσχέτιση μεταξύ του χρόνου εκτέλεσης και του αριθμού εκτελέσεων στο Κβαντικό υλικό όπως και ήταν λογικό.

Έστερα παρουσιάζουμε ένα διάγραμμα που θα μπορούσαμε να πούμε πως διαφαίνεται για πρώτη φορά στα πλαίσια αυτής της εργασίας το Quantum Supremacy σε πραγματικό πείραμα

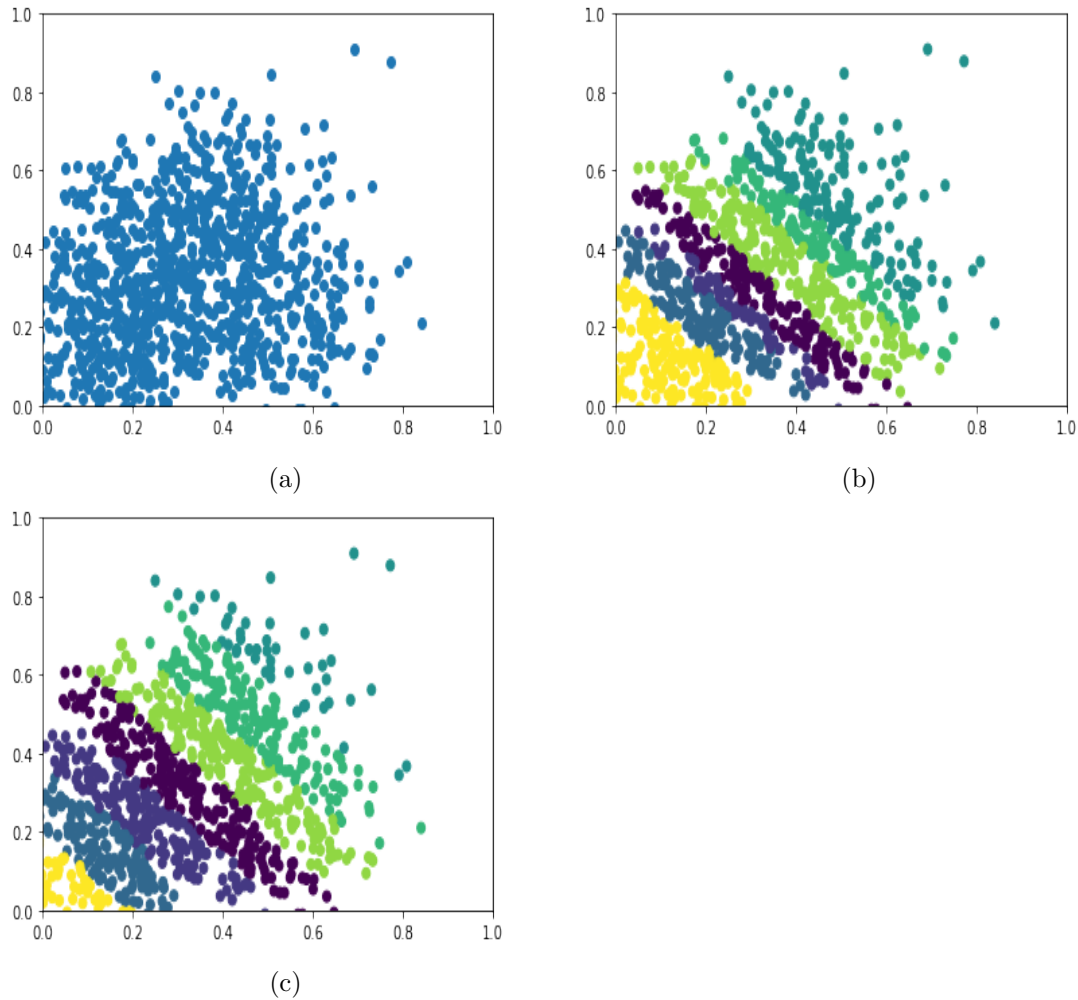


Figure 5.12: Ομαδοποίηση 1000 σημείων προϊόντος του χρόνου

5.16 Δείχνουμε τη συσχέτιση ανάμεσα στο μέγεθος της διάστασης των χαρακτηριστικών της εισόδου και στο χρόνο εκτέλεσης στο Κβαντικό και το Κλασσικό μοντέλο. Επειδή υπάρχει μεγάλη διαφορά στις τάξεις μεγέθους, για το Ctime κάνουμε ένα normalization των τιμών. Παρόλα αυτά αυτό που παρατηρούμε είναι πως ενώ με την αύξηση του μεγέθους της εισόδου στο Κλασσικό μοντέλο έχουμε μεγάλη αύξηση του χρόνου εκτέλεσης, στο αντίστοιχο Κβαντικό δεν παρατηρείται τέτοια αύξηση. Αυτό μας δείχνει, κάτι που γνωρίζαμε ήδη από τη θεωρητική ανάλυση. Ο κβαντικό αλγόριθμος δεν επηρεάζεται τόσο από την είσοδο αφού η πολυπλοκότητά του είναι της τάξεως του  $\log(n)$ .

Τέλος παρουσιάζουμε ένα διάγραμμα που δείχνει τη σχέση του χρόνου εκτέλεσης στο Κβαντικό υλικό με τον αριθμό των εκτελέσεων που χρειαζόμαστε για να πάρουμε ασφαλή αποτελέσματα 5.17. Όπως είναι φυσικό παρατηρούμε πως περισσότερες εκτελέσεις μας οδηγούν σε μεγαλύτερο χρόνο εκτέλεσης.

	Qtime	Ctime	features	shots
0	6.627725	0.000197	2.0	4.0
1	7.054593	0.000147	2.0	8.0
2	6.698065	0.000147	2.0	16.0
3	6.829511	0.000149	2.0	256.0
4	6.858196	0.000152	2.0	512.0

Figure 5.13: Final Dataset

	Qtime	Ctime	features	shots
Qtime	1.000000	0.046473	0.149386	0.324195
Ctime	0.046473	1.000000	0.648112	0.204988
features	0.149386	0.648112	1.000000	0.377049
shots	0.324195	0.204988	0.377049	1.000000

Figure 5.14: Pearson correlation

	X	Y	method	alternative	n	r	CI95%	p-unc	power
0	Qtime	Ctime	spearman	two-sided	102	0.108	[-0.09, 0.3]	0.279	0.192
1	Qtime	features	spearman	two-sided	102	0.262	[0.07, 0.43]	0.008	0.763
2	Qtime	shots	spearman	two-sided	102	0.435	[0.26, 0.58]	0.000	0.997
3	Ctime	features	spearman	two-sided	102	0.357	[0.17, 0.52]	0.000	0.961
4	Ctime	shots	spearman	two-sided	102	0.273	[0.08, 0.44]	0.006	0.799
5	features	shots	spearman	two-sided	102	0.204	[0.01, 0.38]	0.040	0.542

Figure 5.15: Spearman correlation

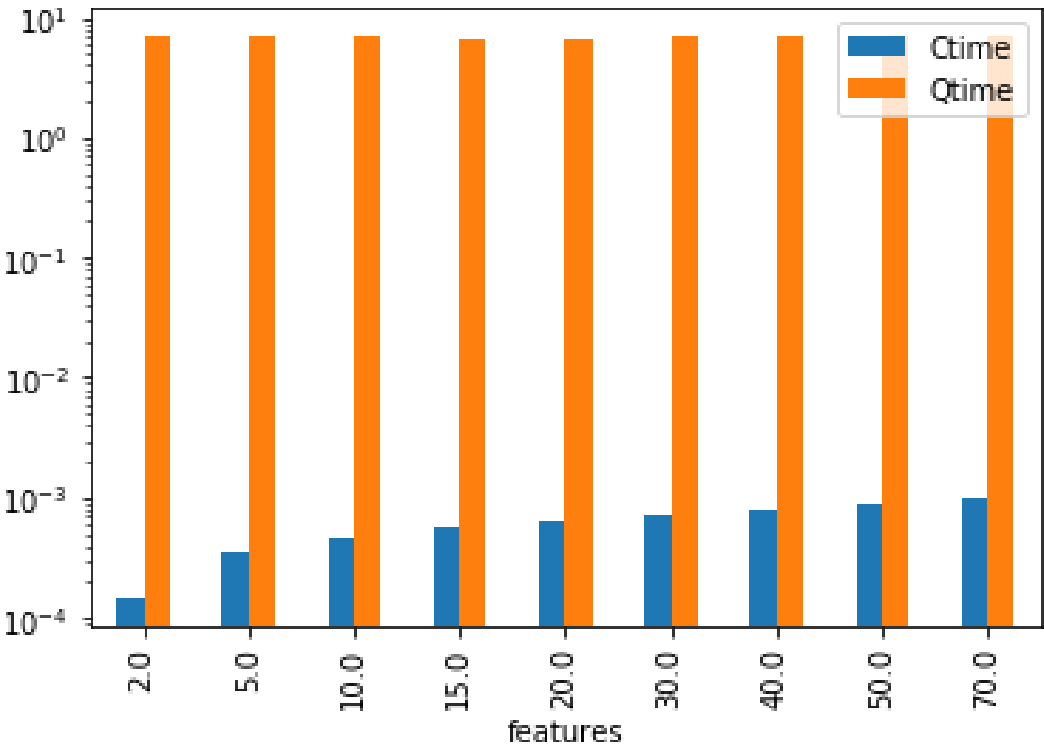


Figure 5.16: Features - Ctime and Qtime

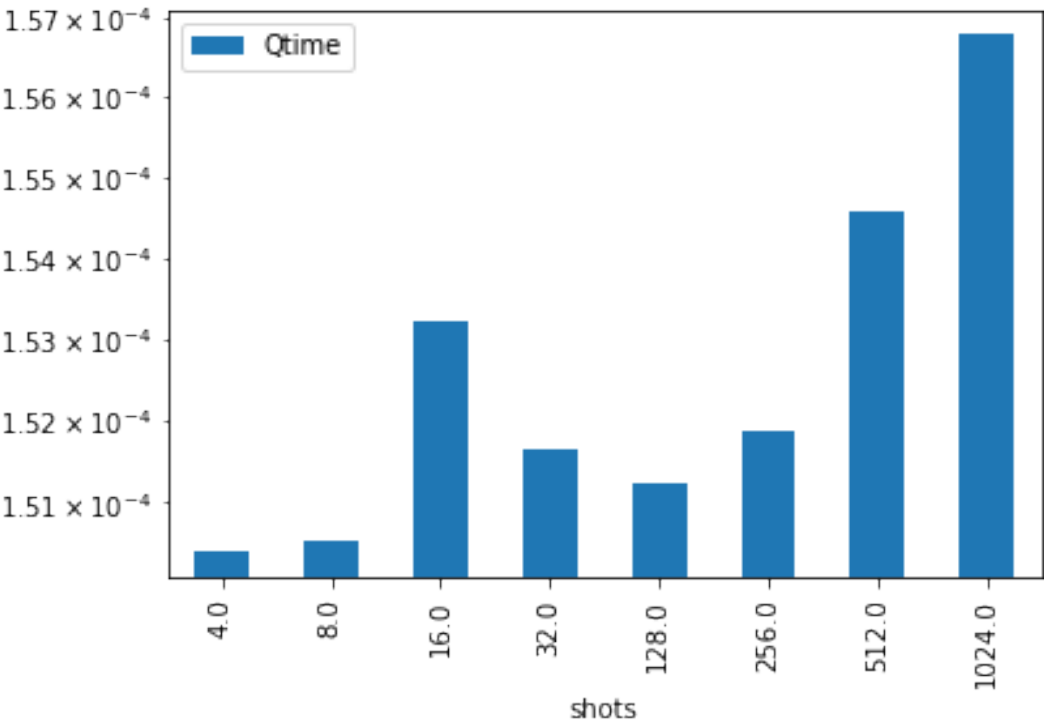


Figure 5.17: Shots and Qtime



## 6

# Επίλογος-Συμπεράσματα

*No More Games. No More Bombs. No More Walking. No More Fun. No More Swimming. 67. That is 17 years past 50. 17 more than I needed or wanted. Boring. I am always bitchy. No Fun – for anybody. 67. You are getting Greedy. Act your old age. Relax – This won't hurt.*

*Hunter Tomson*

## 6.1 Συμπεράσματα

Μπορούμε με ασφάλεια να πούμε πως το Κβαντικό υλικό που είναι διαθέσιμο για ευρεία χρήση και κατανάλωση, ανοιχτό στο κοινό δηλαδή, δεν είναι ακόμα αξιόπιστο σε ό,τι αφορά το χρόνο εκτέλεσης. Μπορούμε επίσης να πούμε με σιγουριά πως η ανωτερότητα των Κβαντικών αλγορίθμων όσον αφορά τη χρονική πολυπλοκότητα διαφαίνεται ακόμα και στην εκτέλεση σε αυτό το υλικό.

Ο κύριος λόγος που βλέπουμε τόσο περισσότερο χρόνο εκτέλεσης στο Κβαντικό υλικό από ό,τι θα περιμέναμε είναι το φαινόμενο decoherence [37]. Παρότι με την τεχνολογική πρόοδο στον τομέα έχουμε καθαρότερα qubits, καλύτερες λογικές πύλες, πιο έξυπνους αλγόριθμους, το φαινόμενο του decoherence αποτελεί τροχοπέδη για να έχουμε ακόμα φθηνούς, προσιτούς σε όλο το κοινό Κβαντικούς υπολογιστές. Η χρήση των κβαντικών λογικών πυλών επιφέρει τον παράγοντα του error στην Κβαντική μέτρηση, εξαιτίας του decoherence, τη διαδικασία δηλαδή όπου το Κβαντικό μοντέλο αναπόφευκτα αλληλεπιδρά με το περιβάλλον και κατά συνέπεια τα qubits βρίσκονται εκτεθειμένα και η κβαντική τους κατάσταση αλλάζει ανεξέλεγκτα και ανεπιθύμητα. Έτσι πληροφορία που αποθηκεύεται στο Κβαντικό υλικό χάνεται.

Εξαιτίας αυτού του φαινομένου εικάζουμε πως βλέπουμε τόσο μεγάλο χρόνο εκτέλεσης στους Κβαντικούς υπολογιστές που είναι ανοιχτοί για το κοινό, καθώς η αποσιωπησή του με φθηνές μεθόδους προσθέτει αναπόφευκτα στο χρόνο εκτέλεσης. Η ύπαρξη του φαινομένου οφείλεται σε πολλούς παράγοντες όπως η ύπαρξη μαγνητικών ή ηλεκτρικών πεδίων στο περιβάλλον, ραδιενέργεια από το περιβάλλον ή ακόμα και προβλήματα στην αλληλεπίδραση μεταξύ των qubits.

Σίγουρα με την επίτευξη σταθερότερων Κβαντικών υπολογιστών για ευρεία κατανάλωση, θα δούμε τρομακτική επιτάχυνση στην εκτέλεση σωρείας αλγορίθμων που χρησιμοποιούμε σήμερα.

Αξίζει να σημειωθεί πως αρκετοί επιστήμονες του κλάδου διατείνονται πως η μέτρηση του χρόνου εκτέλεσης με τη συνάρτηση  $\text{time\_taken}()$  δεν είναι ακριβής και ορθή και χρειάζεται

η δημιουργία μιας system-agnostic μεθόδου που θα δείχνει καλύτερο το speedup για την εκτέλεση σε Κβαντικό και Κλασσικό υλικό. Σε αυτή τη λογική έχει προταθεί η μετρική που βασίζεται στα CLOPS (Circuit Layer Operations Per Second) [38]. Πόση εργασία εκτελεί ένα Κβαντικό σύστημα σε μια συγκεκριμένη μονάδα χρόνου; Η μετρική αυτή είναι άμεσα συνδεδεμένη με το πόσο γρήγορα μπορεί ένα Κβαντικό σύστημα να εκτελέσει συγκεκριμένα ένα Κβαντικό κύκλωμα. Μετράει την ταχύτητα ενός Κβαντικού επεξεργαστή που εκτελεί συγκεκριμένα layers ενός παραμετροποιημένου μοντέλου κυκλώματος.

Στα CLOPS και το μέγεθός τους μετέχουν αρκετά aspects του Κβαντικού hardware και software όπως:

- Ο βαθμός επαναληψιμότητας μιας διεργασίας στον Κβαντικό επεξεργαστή
- Η ταχύτητα με την οποία τρέχουν οι λογικές πύλες
- Ο χρόνος που παίρνει το compilation στο runtime
- Ο χρόνος που χρειάζεται για να παραχθούν οι Κλασσικές εντολές ελέγχου
- Ο χρόνος που παίρνει για να μεταφερθεί η πληροφορία και τα δεδομένα ανάμεσα στα Κβαντικά units

## 6.2 Future work

Σαν συνέχεια αυτής της εργασίας προτείνεται

- Η βαθύτερη μελέτη του υβριδικού kmeans που προτείναμε ώστε
  - Περισσότερα layers του αλγορίθμου να λειτουργούν σε Κβαντικό επίπεδο
  - Να μπορεί να δεχθεί είσοδο μεγαλύτερων διαστάσεων, το οποίο θα επιτευχθεί με Κβαντική αναπαράσταση μεγαλύτερων κλασσικών states [39]
- Το να αποκτήσουμε πρόσβαση για να τρέχουμε σε μεγαλύτερους χώρους καταστάσεων, χρησιμοποιώντας περισσότερα από 5 qubits, πχ στο πρόγραμμα IBM Quantum Researchers Program, ειδικά για ερευνητές και ιδρύματα, όπου μπορούμε να έχουμε πρόσβαση και στο microwave pulse control για qubits
- Το πάντρεμα των δύο αλγορίθμων που υλοποιήσαμε σε ένα πρόβλημα όπως Time Series Clustering
- Μελέτη των CLOPS και μέτρηση του χρόνου εκτέλεσης χρησιμοποιώντας αυτά σαν μονάδα μέτρησης
- Μελέτη Κβαντικών Αλγορίθμων για Νευρωνικά Δίκτυα και συγκεκριμένα του Bayesian Deep Learning
- Συνδιασμός των Κβαντικών Νευρωνικών Δικτύων για Deep Reinforcement Learning, με Double Deep Q Networks στα πρότυπα της προηγούμενης δουλειάς μας στο [40] [41]



*If quantum theory is correct, it signifies the end of physics as a science.*

*Albert Einstein*

*Einstein said that if quantum mechanics were correct then the world would be  
crazy. Einstein was right - the world is crazy*

*Daniel M. Greenberger*



# Bibliography

- [1] Panu Raatikainen. Gödel’s Incompleteness Theorems. In Edward N. Zalta, editor, *The Stanford Encyclopedia of Philosophy*. Metaphysics Research Lab, Stanford University, Spring 2022 edition, 2022.
- [2] Andrew Warwick. Cambridge mathematics and cavendish physics: Cunningham, campbell and einstein’s relativity 1905?1911 part i: The uses of theory. *Studies in History and Philosophy of Science Part A*, 23(4):625–656, 1991.
- [3] Gregg Jaeger. *Double-Slit Experiment (or Two-Slit Experiment)*, pages 174–178. Springer Berlin Heidelberg, Berlin, Heidelberg, 2009.
- [4] George Temple. *The General Principles of Quantum Theory*. Routledge, 2014.
- [5] Michael A. Nielsen and Isaac L. Chuang. *Quantum Computation and Quantum Information: 10th Anniversary Edition*. Cambridge University Press, USA, 10th edition, 2011.
- [6] Peter W. Shor. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM Journal on Computing*, 26(5):1484–1509, oct 1997.
- [7] Lov K. Grover. A fast quantum mechanical algorithm for database search, 1996.
- [8] Frank Arute, Kunal Arya, Ryan Babbush, Dave Bacon, Joseph C. Bardin, Rami Barends, Rupak Biswas, Sergio Boixo, Fernando G. S. L. Brandao, David A. Buell, Brian Burkett, Yu Chen, Zijun Chen, Ben Chiaro, Roberto Collins, William Courtney, Andrew Dunsworth, Edward Farhi, Brooks Foxen, Austin Fowler, Craig Gidney, Marissa Giustina, Rob Graff, Keith Guerin, Steve Habegger, Matthew P. Harrigan, Michael J. Hartmann, Alan Ho, Markus Hoffmann, Trent Huang, Travis S. Humble, Sergei V. Isakov, Evan Jeffrey, Zhang Jiang, Dvir Kafri, Kostyantyn Kechedzhi, Julian Kelly, Paul V. Klimov, Sergey Knysh, Alexander Korotkov, Fedor Kostritsa, David Landhuis, Mike Lindmark, Erik Lucero, Dmitry Lyakh, Salvatore Mandrà, Jarrod R. McClean, Matthew McEwen, Anthony Megrant, Xiao Mi, Kristel Michielsen, Masoud Mohseni, Josh Mutus, Ofer Naaman, Matthew Neeley, Charles Neill, Murphy Yuezhen Niu, Eric Ostby, Andre Petukhov, John C. Platt, Chris Quintana, Eleanor G. Rieffel, Pedram Roushan, Nicholas C. Rubin, Daniel Sank, Kevin J. Satzinger, Vadim Smelyanskiy, Kevin J. Sung, Matthew D. Trevithick, Amit Vainsencher, Benjamin Villalonga, Theodore White, Z. Jamie Yao, Ping Yeh, Adam Zalcman, Hartmut Neven, and John M. Martinis. Quantum supremacy using a programmable superconducting processor. *Nature*, 574(7779):505–510, Oct 2019.

- [9] A. M. J. Zwerver, T. Krähenmann, T. F. Watson, L. Lampert, H. C. George, R. Pillarisetty, S. A. Bojarski, P. Amin, S. V. Amitonov, J. M. Boter, R. Caudillo, D. Correias-Serrano, J. P. Dehollain, G. Droulers, E. M. Henry, R. Kotlyar, M. Lodari, F. Lüthi, D. J. Michalak, B. K. Mueller, S. Neyens, J. Roberts, N. Samkharadze, G. Zheng, O. K. Zietz, G. Scappucci, M. Veldhorst, L. M. K. Vandersypen, and J. S. Clarke. Qubits made by advanced semiconductor manufacturing. *Nature Electronics*, 5(3):184–190, mar 2022.
- [10] Charles T. Sebens. How electrons spin. *Studies in History and Philosophy of Science Part B: Studies in History and Philosophy of Modern Physics*, 68:40–50, nov 2019.
- [11] Anne-Catherine de la Hamette, Viktoria Kabel, Esteban Castro-Ruiz, and Časlav Brukner. Falling through masses in superposition: quantum reference frames for indefinite metrics, 2021.
- [12] Chu-Ryang Wie. Bloch sphere model for two-qubit pure states, 2014.
- [13] François Gieres. Mathematical surprises and dirac's formalism in quantum mechanics. *Reports on Progress in Physics*, 63(12):1893–1931, nov 2000.
- [14] Quantum propability and measurement.
- [15] Ryszard Horodecki, Paweł Horodecki, Michał Horodecki, and Karol Horodecki. Quantum entanglement. *Reviews of Modern Physics*, 81(2):865–942, jun 2009.
- [16] Lin Weng. Local and global quantum gates, 2021.
- [17] Most famous quantum gates.
- [18] Dorit Aharonov, Alexei Kitaev, and Noam Nisan. Quantum circuits with mixed states. In *Proceedings of the Thirtieth Annual ACM Symposium on Theory of Computing*, STOC '98, page 20–30, New York, NY, USA, 1998. Association for Computing Machinery.
- [19] Adam Bouland. *Space around bounded-error quantum polynomial time*. PhD thesis, 01 2017.
- [20] Christos H. Papadimitriou. *Computational Complexity*, page 260–265. John Wiley and Sons Ltd., GBR, 2003.
- [21] Adi Shamir.  $\text{Ip} = \text{pspace}$ . *J. ACM*, 39(4):869–877, oct 1992.
- [22] Liyi Li, Finn Voichick, Kesha Hietala, Yuxiang Peng, Xiaodi Wu, and Michael Hicks. Verified compilation of quantum oracles, 2021.
- [23] Bo Moon. The subset sum problem: Reducing time complexity of np-completeness with quantum search. *Undergraduate Journal of Mathematical Modeling: One + Two*, 4, 05 2013.
- [24] Daan Camps, Roel Van Beeumen, and Chao Yang. Quantum fourier transform revisited. *Numerical Linear Algebra with Applications*, 28(1), sep 2020.
- [25] C. Candan, M.A. Kutay, and H.M. Ozaktas. The discrete fractional fourier transform. *IEEE Transactions on Signal Processing*, 48(5):1329–1337, 2000.

- [26] External Advisor, Leonard Schulman, Internal Advisor, Piyush, P Kurur, and Abhishek Bhowmick. A theoretical analysis of lloyd’s algorithm for k-means clustering. 04 2022.
- [27] Dawid Kopczyk.
- [28] Min-Sung Kang, Jino Heo, Seong-Gon Choi, Sung Moon, and Sang-Wook Han. Implementation of swap test for two unknown states in photons via cross-kerr nonlinearities under decoherence effect. *Scientific reports*, 9(1):6167–6167, Apr 2019. 30992536[pmid].
- [29] Paul W. Mielke and Kenneth J. Berry. *Euclidean Distance Based Permutation Methods in Atmospheric Science*, pages 7–27. Springer US, Boston, MA, 2000.
- [30] Encoding of classical data to quantum. *Towards Data Science*.
- [31] Ibm quantum experience.
- [32] Ibm.
- [33] Qiskit terra.
- [34] Qiskit aer.
- [35] Quantum providers.
- [36] Joseph Lee Rodgers and W. Alan Nicewander. Thirteen ways to look at the correlation coefficient. *The American Statistician*, 42(1):59–66, 1988.
- [37] Encoding of classical data to quantum. *Towards Data Science*.
- [38] Andrew Wack, Hanhee Paik, Ali Javadi-Abhari, Petar Jurcevic, Ismael Faro, Jay M. Gambetta, and Blake R. Johnson. Quality, speed, and scale: three key attributes to measure the performance of near-term quantum computers, 2021.
- [39] Steven Herbert. Increasing the classical data throughput in quantum networks by combining quantum linear network coding with superdense coding. *Phys. Rev. A*, 101:062332, Jun 2020.
- [40] Constantinos Bitsakos, Ioannis Konstantinou, and Nectarios Koziris. Derp: A deep reinforcement learning cloud system for elastic resource provisioning. In *2018 IEEE International Conference on Cloud Computing Technology and Science (CloudCom)*, pages 21–29, 2018.
- [41] Zhikuan Zhao, Alejandro Pozas-Kerstjens, Patrick Rebentrost, and Peter Wittek. Bayesian deep learning on a quantum computer. *Quantum Machine Intelligence*, 1(1-2):41–51, may 2019.