IRQ: Introduction to Linear Algebra

Dr. Kamalika Datta

Dr. Abhoy Kole

M.Sc. Lennart Weingarten

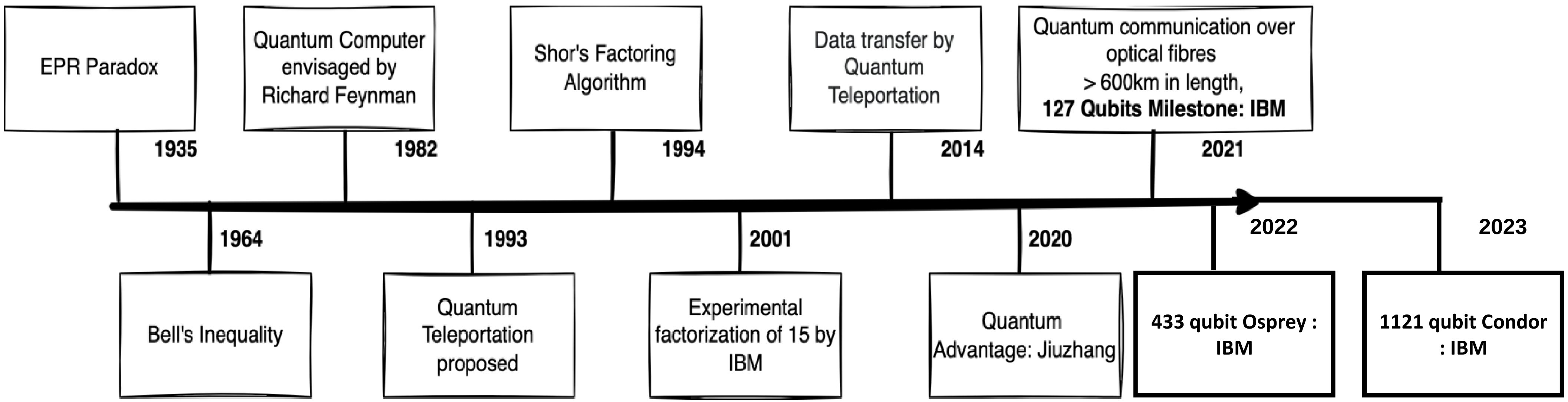
Lecture - 1

WINTER SEMESTER 2024/2025

Lecture Coverage

- Motivation
- Recent Trend
- Introduction to Quantum Computing
- Complex Numbers
- Quantum State representation

Timeline Exploring the Development of Quantum Technologies



Ayoade, O.; Rivas, P.; Orduz, J. Artificial Intelligence Computing at the Quantum Level. Data 2022, 7, 28. https://doi.org/ 10.3390/data7030028

Motivation

- 1982: Simulating Physics with Computers
- Rich

"Nature isn't classical, dammit, and if you want to make a simulation of nature, you'd better make it quantum mechanical, and by golly it's a wonderful problem, because it doesn't look so easy"

--- Richard P. Feynman, 1981

R. P. Feynman, "Simulating physics with computers", Int'l Journal of Theoretical Physics, vol. 21, pp. 467–488, 1982
R. P. Feynman, "Quantum mechanical computers", Optic News, vol. 11, pp. 11-20, 1985.

Motivation

- 1994: Polynomial time algorithm for factoring large numbers on a quantum computer
- Peter Shor
- Bell Labs

SHOR's FACTORIZATION ALGORITHM

MAJOR BREAKTHOUGH IN QUANTUM COMPUTING

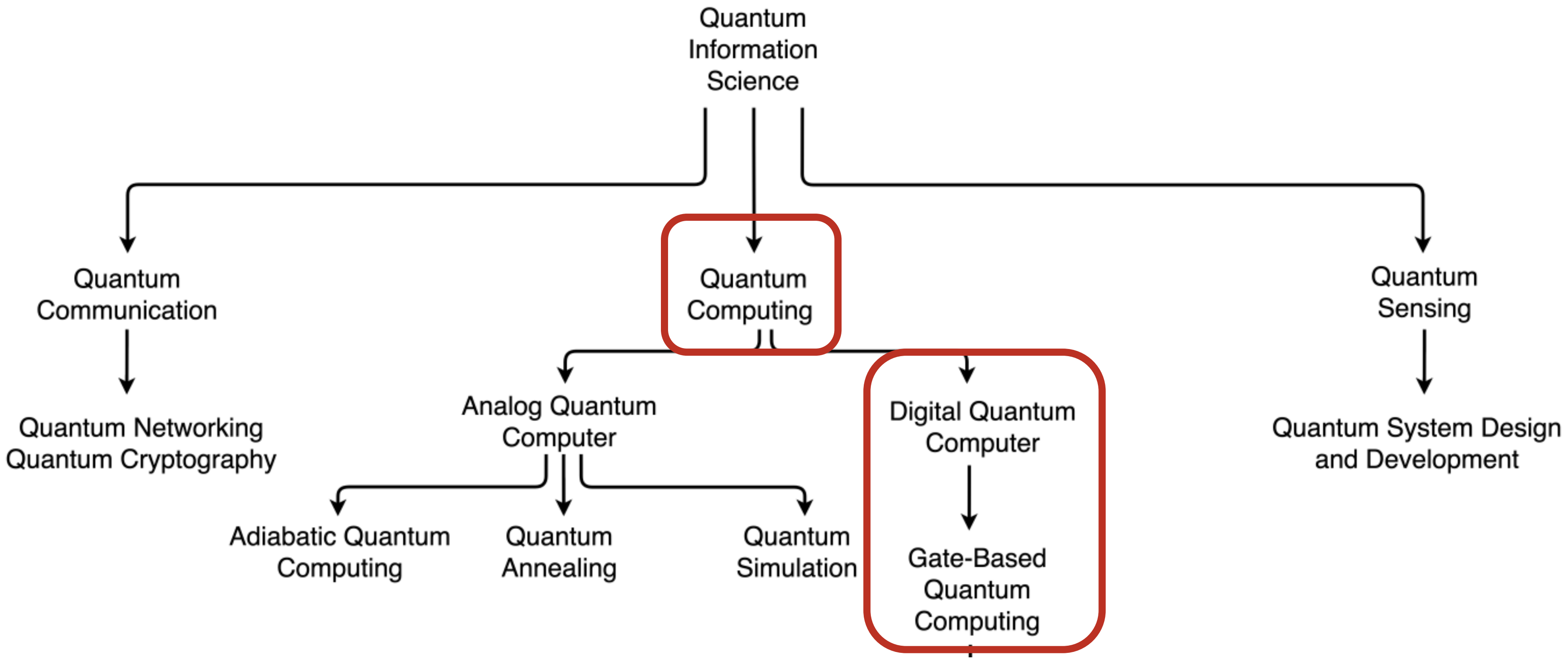
P. W. Shor, Algorithms for quantum computation: Discrete logarithms and factoring. In Symp. on Foundations of Computer Science. 124–134, 1994

Motivation

- 1996: Quantum database search algorithm
- Quadratic speedup
- Lov Grover
- Bell Labs

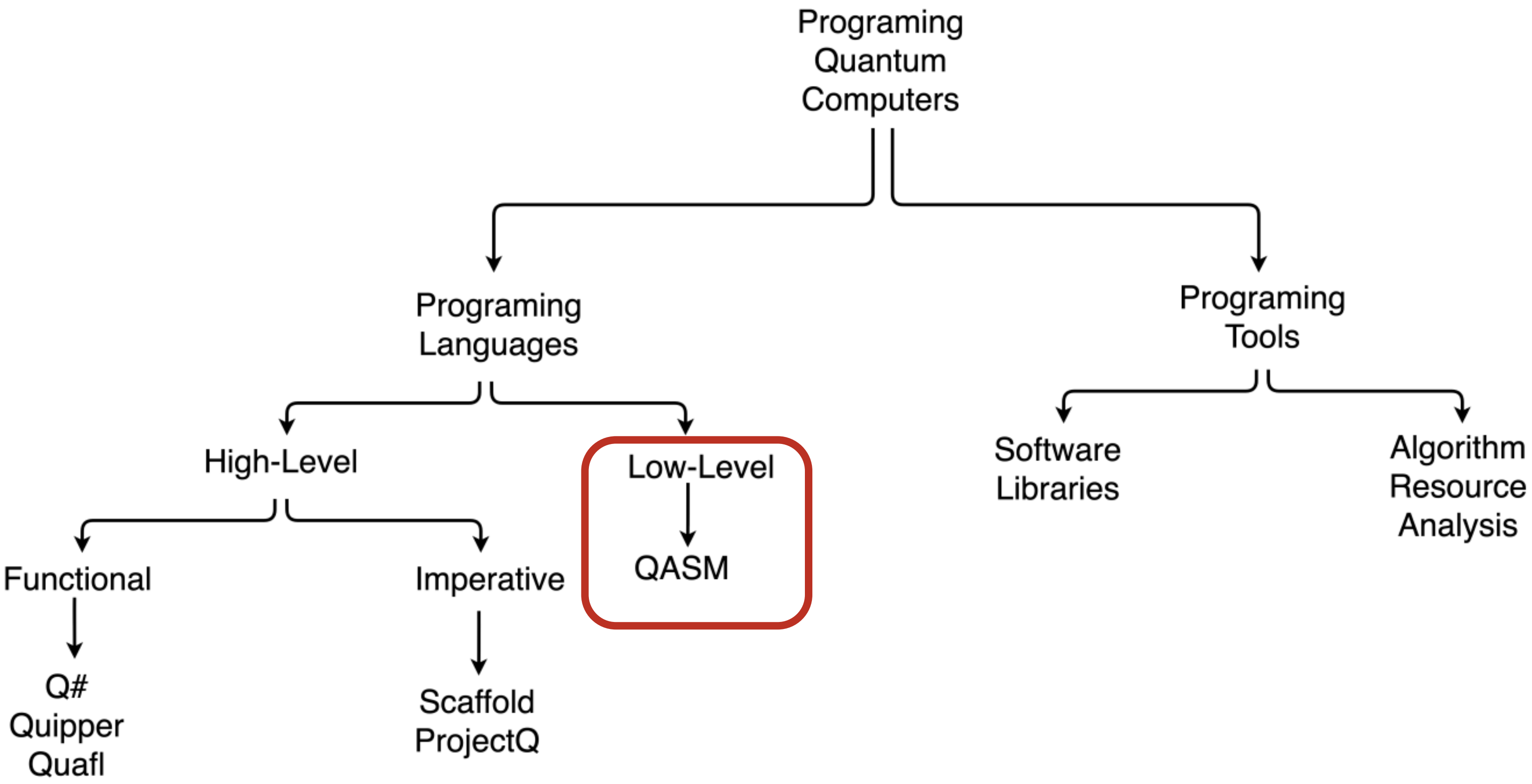
L. Grover, "A fast quantum mechanical algorithm for database search," in ACM Symp. on the Theory of Computing, p. 212, 1996

Quantum Information Science Chart



Ayoade, O.; Rivas, P.; Orduz, J. Artificial Intelligence Computing at the Quantum Level. Data 2022, 7, 28. https://doi.org/ 10.3390/data7030028

Programming Quantum Computers Chart



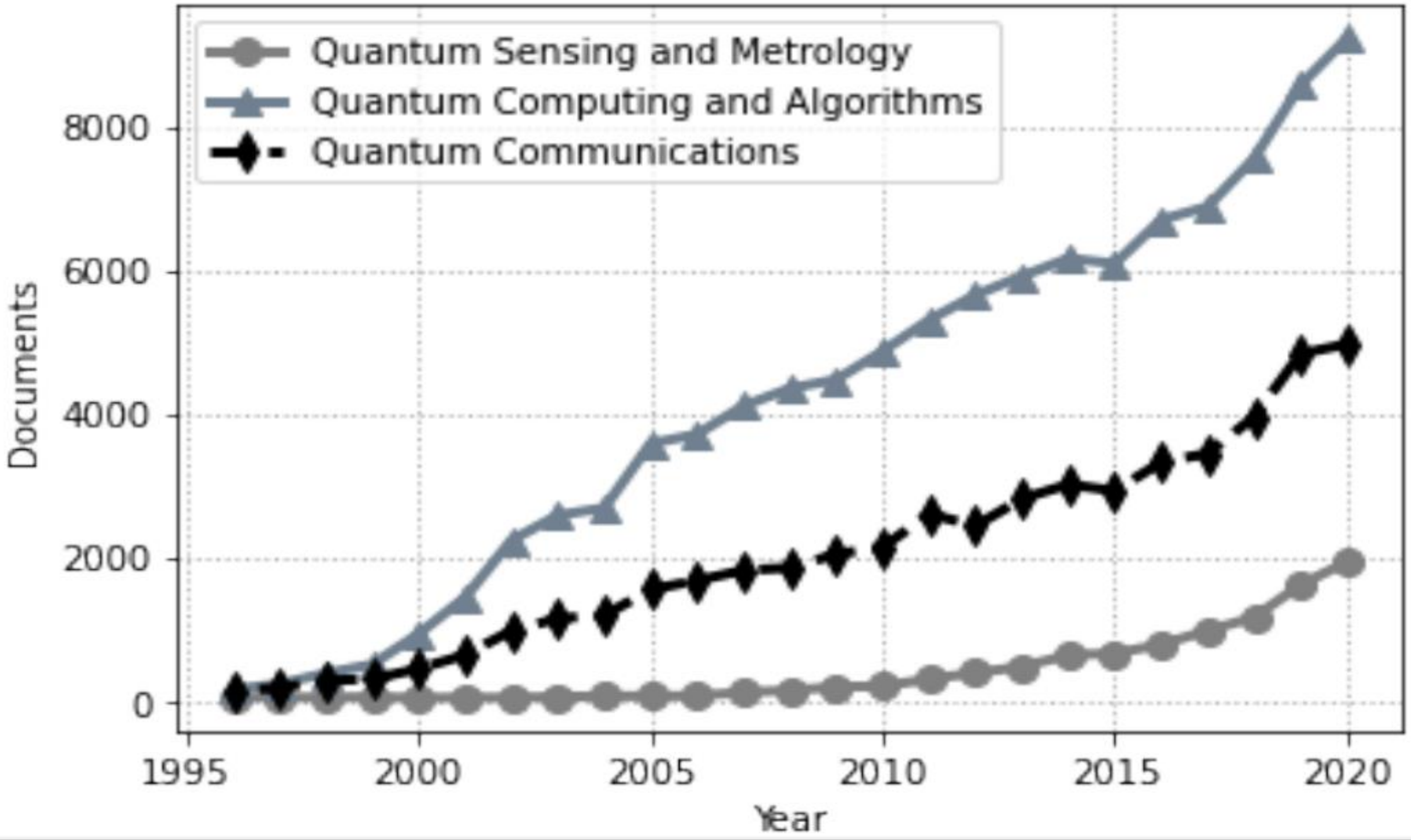
Ayoade, O.; Rivas, P.; Orduz, J. Artificial Intelligence Computing at the Quantum Level. Data 2022, 7, 28. https://doi.org/ 10.3390/data7030028

Quantum Computing Technologies

Trapped Ion Qubit	Superconducting Qubit	Photonic Qubit
To produce qubits, lasers are used to ionize atoms and trap them in electric potentials. The status of the qubits is then measured using an extra laser.	The qubits are created by combining a superconducting resonator with a nonlinear inductor to make an artificial atom.	The squeezed state (light working as qubit) is created by distributing laser light to an array of squeezers (microscopic devices comprised of relatively small ring resonators) .
Stable qubits can be generated using trapped ion technology, and forming an entangled state is simple. Working with large numbers of qubits in this system is challenging, and implementing a whole quantum algorithm is even more complicated. Decoherence is a difficult problem to solve.	Building and accurately measuring qubits with superconducting technology is simple. These qubits have a nanosecond time scale and a quick decoherence time. Qubits must be cooled to near absolute zero to function, and computation is subject to quantum noise.	Qubits are far more stable in photonic technology and can readily entangle a huge number of photons. It is possible to perform computation at room temperature, but it is less fault-tolerant, and error correction is harder. According to this technique, quantum supremacy is attained.

Ayoade, O.; Rivas, P.; Orduz, J. Artificial Intelligence Computing at the Quantum Level. Data 2022, 7, 28. https://doi.org/ 10.3390/data7030028

Quantum Information Science Progress Report



Ayoade, O.; Rivas, P.; Orduz, J. Artificial Intelligence Computing at the Quantum Level. Data 2022, 7, 28. https://doi.org/ 10.3390/data7030028

Industry Players

IBM Q

Microsoft

IONQ

Recent Trends

- Launch of IBM's first Quantum Computer in May 2016.

QISKIT by IBM's research March 2017

IBM's first Quantum Computer May 2016 (5-qubit)

CANARY

IBM's 16-qubit 2017

Falcon

IBM's 27-qubit 2018

Egret

IBM's 33-qubit

Hummingbird

IBM's 65-qubit 2020

Eagle

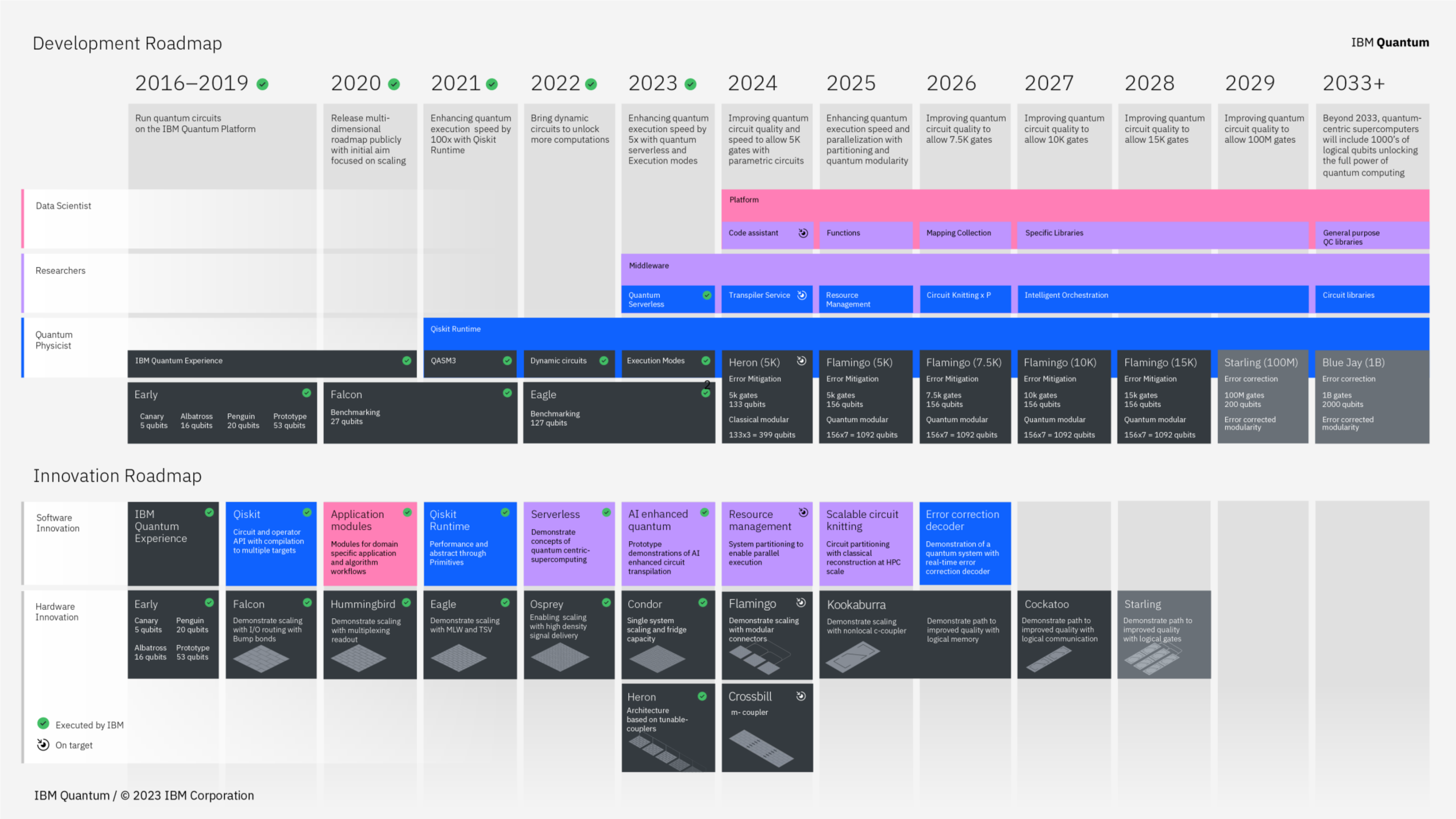
IBM's 127-qubit 2021

Osprey

IBM's 433-qubit 2022

Condor

IBM's 1121-qubit 2023


Development Roadmap
IBM Quantum
2016–2019
2020
2021
2022
2023
2024
2025
2026
2027
2028
2029
2033+
Run quantum circuits on the IBM Quantum Platform
Release multi-dimensional roadmap publicly with initial aim focused on scaling
Enhancing quantum execution speed by 100x with Qiskit Runtime
Bring dynamic circuits to unlock more computations
Enhancing quantum execution speed by 5x with quantum serverless and Execution modes
Improving quantum circuit quality and speed to allow 5K gates with parametric circuits
Enhancing quantum execution speed and parallelization with partitioning and quantum modularity
Improving quantum circuit quality to allow 7.5K gates
Improving quantum circuit quality to allow 10K gates
Improving quantum circuit quality to allow 15K gates
Improving quantum circuit quality to allow 100M gates
Beyond 2033, quantum-centric supercomputers will include 1000's of logical qubits unlocking the full power of quantum computing
Data Scientist
Platform
Code assistant
Functions
Mapping Collection
Specific Libraries
General purpose QC libraries
Researchers
Middleware
Quantum Serverless
Transpiler Service
Resource Management
Circuit Knitting x P
Intelligent Orchestration
Circuit libraries
Quantum Physicist
Qiskit Runtime
IBM Quantum Experience
QASM3
Dynamic circuits
Execution Modes
Heron (5K) Error Mitigation 5k gates 133 qubits Classical modular 133x3 = 399 qubits
Flamingo (5K) Error Mitigation 5k gates 156 qubits Quantum modular 156x7 = 1092 qubits
Flamingo (7.5K) Error Mitigation 7.5k gates 156 qubits Quantum modular 156x7 = 1092 qubits
Flamingo (10K) Error Mitigation 10k gates 156 qubits Quantum modular 156x7 = 1092 qubits
Flamingo (15K) Error Mitigation 15k gates 156 qubits Quantum modular 156x7 = 1092 qubits
Starling (100M) Error correction 100M gates 200 qubits Error corrected modularity
Blue Jay (1B) Error correction 1B gates 2000 qubits Error corrected modularity
Early Canary 5 qubits Albatross 16 qubits Penguin 20 qubits Prototype 53 qubits
Falcon Benchmarking 27 qubits
Eagle Benchmarking 127 qubits
Innovation Roadmap
Software Innovation
IBM Quantum Experience
Qiskit Circuit and operator API with compilation to multiple targets
Application modules Modules for domain specific application and algorithm workflows
Qiskit Runtime Performance and abstract through Primitives
Serverless Demonstrate concepts of quantum centric-supercomputing
AI enhanced quantum Prototype demonstrations of AI enhanced circuit transpilation
Resource management System partitioning to enable parallel execution
Scalable circuit knitting Circuit partitioning with classical reconstruction at HPC scale
Error correction decoder Demonstration of a quantum system with real-time error correction decoder
Hardware Innovation
Early Canary 5 qubits Penguin 20 qubits Albatross 16 qubits Prototype 53 qubits
Falcon Demonstrate scaling with I/O routing with Bump bonds
Hummingbird Demonstrate scaling with multiplexing readout
Eagle Demonstrate scaling with MLW and TSV
Osprey Enabling scaling with high density signal delivery
Condor Single system scaling and fridge capacity
Flamingo Demonstrate scaling with modular connectors
Kookaburra Demonstrate scaling with nonlocal c-coupler
Demonstrate path to improved quality with logical memory
Cockatoo Demonstrate path to improved quality with logical communication
Starling Demonstrate path to improved quality with logical gates
Heron Architecture based on tunable-couplers
Crossbill m- coupler
Executed by IBM
On target
IBM Quantum / © 2023 IBM Corporation

Milestones Achieved

- Solution to some computationally harder problems possible.
- We have small scale physical quantum computers.
- Quantum supremacy using a programmable superconducting processor – *Nature 574 505-510 (2019)*.

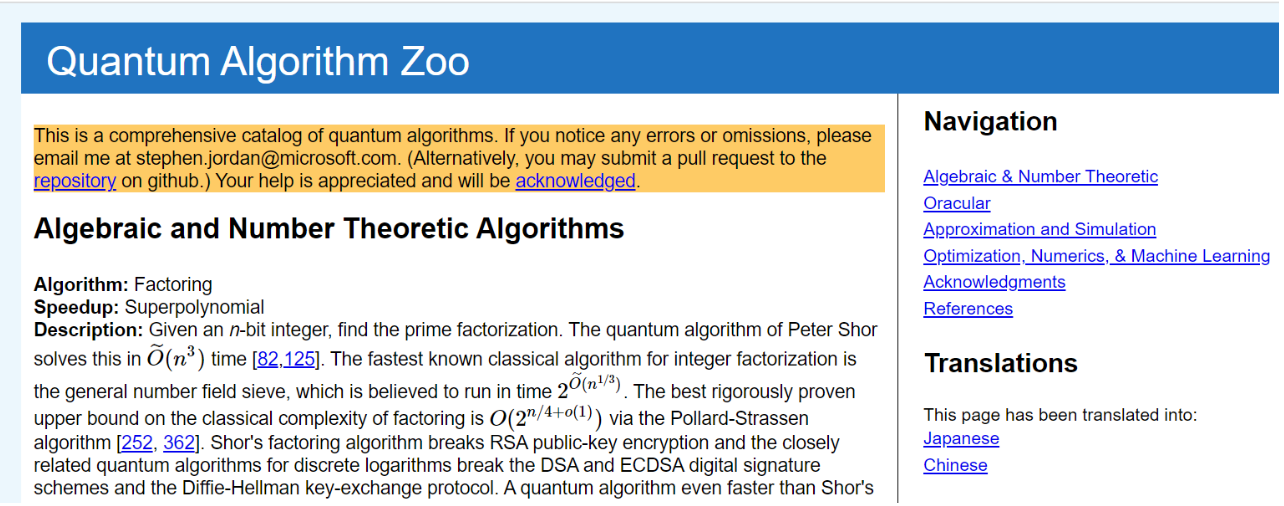
Future Directions

- Making reliable quantum computers.
  - Fault Tolerant Computing
- Design of more quantum algorithms for various applications.
- Some important applications: Quantum Machine Learning (QML), Quantum Cryptography (QC), optimization problems.

Comprehensive Catalogue of Quantum Algorithms

- https://quantumalgorithmzoo.org/

- All known quantum algorithms are listed with their expected speedup over classical computing



Quantum Algorithm Zoo

This is a comprehensive catalog of quantum algorithms. If you notice any errors or omissions, please email me at stephen.jordan@microsoft.com. (Alternatively, you may submit a pull request to the repository on github.) Your help is appreciated and will be acknowledged.

Algebraic and Number Theoretic Algorithms

Algorithm: Factoring
Speedup: Superpolynomial
Description: Given an *n*-bit integer, find the prime factorization. The quantum algorithm of Peter Shor solves this in $\widetilde{O}(n^3)$ time [82,125]. The fastest known classical algorithm for integer factorization is the general number field sieve, which is believed to run in time $2^{\widetilde{O}(n^{1/3})}$. The best rigorously proven upper bound on the classical complexity of factoring is $O(2^{n/4+o(1)})$ via the Pollard-Strassen algorithm [252, 362]. Shor's factoring algorithm breaks RSA public-key encryption and the closely related quantum algorithms for discrete logarithms break the DSA and ECDSA digital signature schemes and the Diffie-Hellman key-exchange protocol. A quantum algorithm even faster than Shor's

Navigation

Algebraic & Number Theoretic
Oracular
Approximation and Simulation
Optimization, Numerics, & Machine Learning
Acknowledgments
References

Translations

This page has been translated into:
Japanese
Chinese

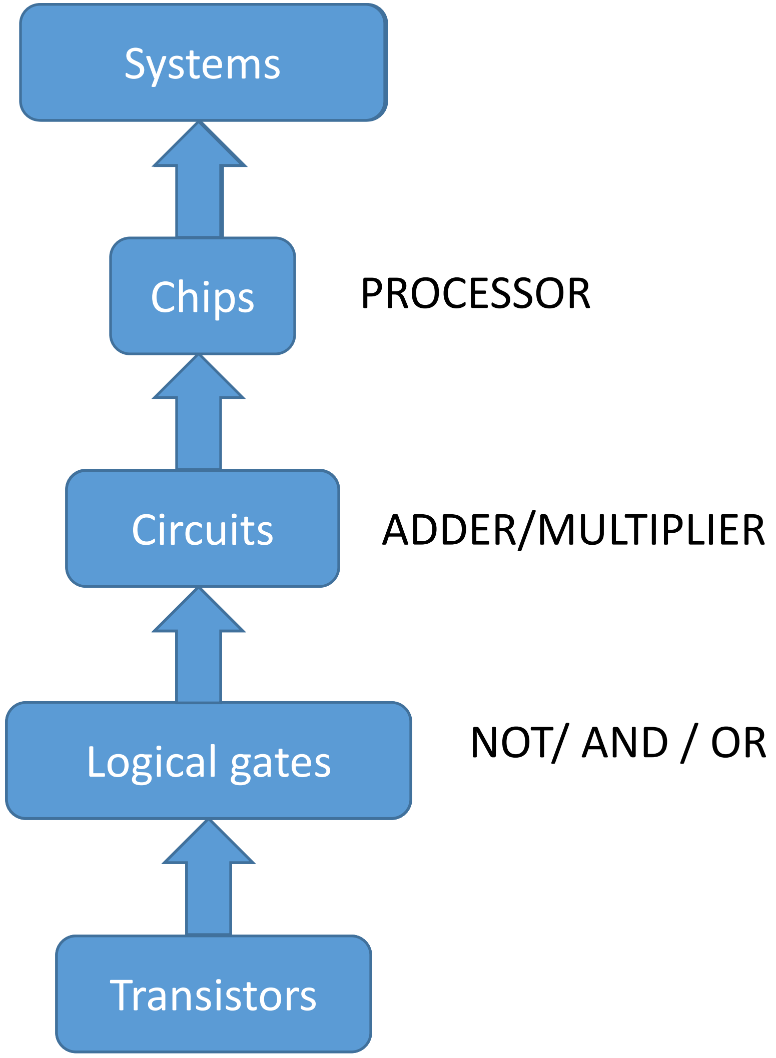
INTRODUCTION TO QUANTUM COMPUTING

Quantum Computing

- A new computing paradigm based on quantum mechanical principles.
- Significantly different from classical computing.
- Qubits are the hardware on which gate operations are carried out sequentially
- Quantum algorithms exist for some problems.
  - Can provide significant speedup (super-linear or exponential).
  - Algorithms expressed as a sequence of quantum gate operations.

Classical Computing

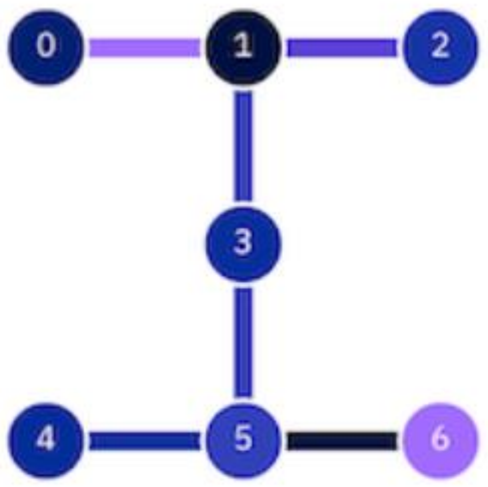
- We carry out computation on basic unit of information called bits.
  - Typically two-valued, 0 and 1.
- We build circuits (e.g. CMOS) that operate on data.
  - Adders, multipliers, etc.
  - Built using logic gates / transistors.



Quantum Computing

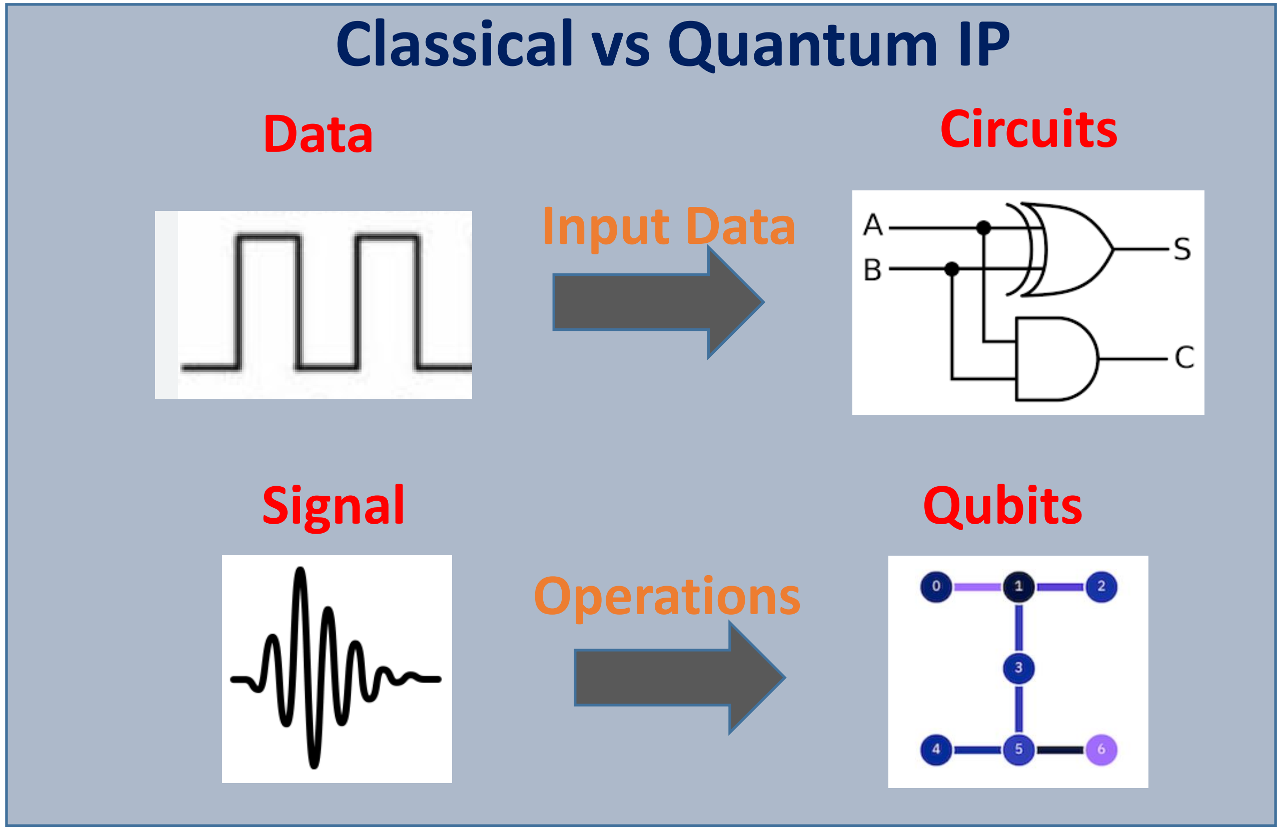
- Basic unit of information is called quantum bits or qubits.
  - Quantum gates operate on qubits to change their states.
  - The qubits are considered as “hardware”, on which the gate operations are performed by applying external stimulus.
- Qubits can exist in state of superposition of the basis states.

$$|\varphi\rangle = \alpha\,|0\rangle + \beta\,|1\rangle$$



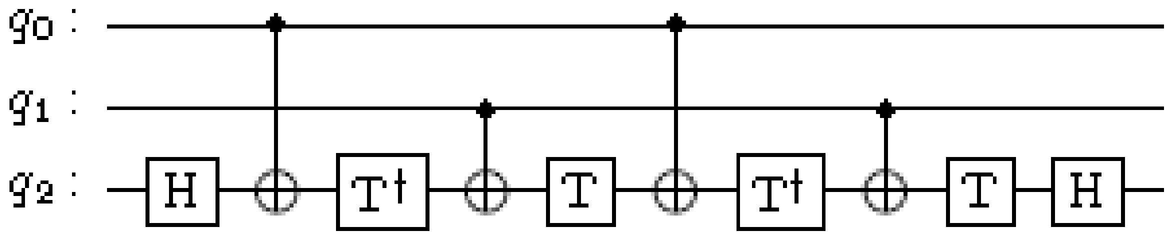
7-qubit FALCON PROCESSOR

Classical and Quantum Information Processing



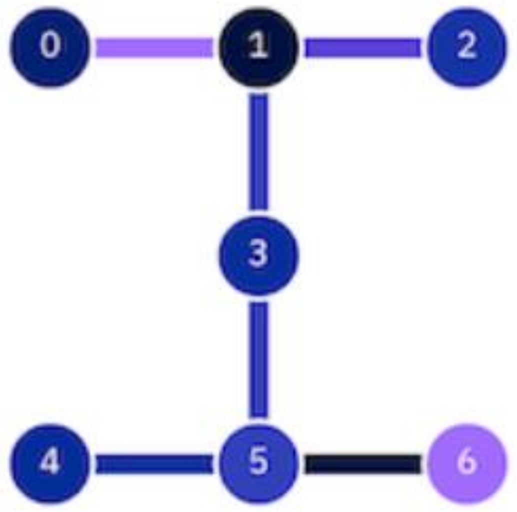
Quantum Computing

- A quantum circuit consists of a sequence of quantum gates.
  - Gate operations are carried out in sequence.
- The execution time depends on the depth of the circuit – that is, number of steps required to carry out the operations.
- Main characteristics of quantum operations:
  - A qubit can exist in state of superposition.
  - Multiple qubits can exist in states of entanglement.



Quantum operations are inherently reversible

Advancement in Quantum Processors



Gate Error rate
Decoherence

IBMQ:	5 qubits (May 2016)
Canary Family:	5 qubits (January 2017)
	16 qubits (May 2017)
Falcon Family:	27 qubits (2019)
Egret Family:	33 qubits
Hummingbird Family:	65 qubits (2020)
Eagle Family:	127 qubits (December 2021)
Osprey Family:	433 qubits (December 2022)
Condor Family:	1121 qubits (December 2023)

https://www.ibm.com/quantum/technology

Classical Vs Quantum information processing

- **Classical operation**
  - AND / OR / NAND / NOR etc
  - Boolean logic
  - State: Scalar (0 / 1)
- **Quantum operation**
  - Hadamard / Pauli X,Y,Z / T / S
  - Linear algebra: Matrix
  - State: vector ($\begin{bmatrix} a1 \\ a2 \end{bmatrix}$)

Necessity of basic Linear Algebra

- To understand the quantum states
- To understand the quantum gate operations
- To understand quantum circuits
- To understand quantum algorithms

What do we need to know?

- Algebra of complex numbers
- Vector or Matrix representation of Quantum State
- Matrix representation of quantum gate operations
- The important quantum gates and their matrix representations
- State transformation

PRELIMINARIES OF QUANTUM COMPUTING

Quantum Information

- A quantum state of a system is represented by a column vector
  - Entries are complex numbers
  - The sum of the absolute values squared of the entries must be 1
  - This is the simplified description
- A quantum state can be represented by general class of matrix known as Density Matrix
  - This includes simplified description and classical information including probabilistic states as special case

Quantum State

- A quantum state $|\varphi\rangle$ can be expressed as the superposition (linear combination) of the basis states.

$$|\varphi\rangle = \alpha\,|0\rangle + \beta\,|1\rangle \qquad |0\rangle = \begin{bmatrix}1\\0\end{bmatrix} \qquad |1\rangle = \begin{bmatrix}0\\1\end{bmatrix}$$

Here $|0\rangle$ **and** $|1\rangle$ are the basis states, and α and β are **complex numbers**

$$|\alpha|^2 + |\beta|^2 = 1$$

$|\varphi\rangle$ can be represented as a vector

Complex Numbers: A revision

COMPLEX NUMBERS

- Complex number representation:

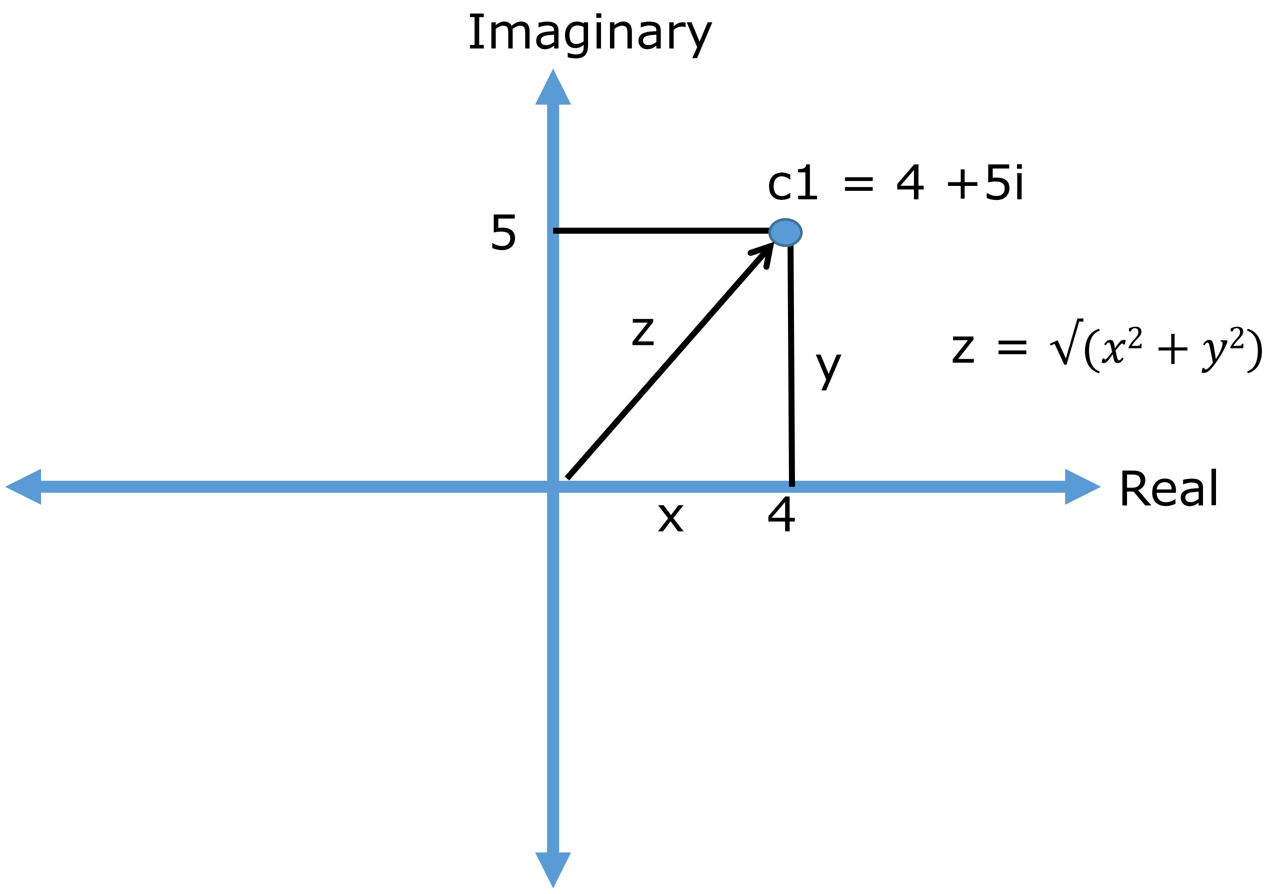
  (A + Bi)

  A – Real part , B – Imaginary part

  i = $\sqrt{-1}$; $i^2 = -1$

- Complex numbers can be added, subtracted, multiplied, divided

Geometry of Complex Numbers



Complex plane / Argand plane

Modulus of a complex number (c1 = a + bi):
Length(c1) = $\sqrt{(a^2 + b^2)}$

Add and Subtract

- Example 1:
  - c1 = 4 +5i, c2 = -8 +2i
- Add c1 and c2
  - Result = c1 + c2 = 4 +5i + -8 +2i = -4 + 7i
- Subtract c2 from c1
  - Result = c1 – c2 = (4 +5i) – (-8 +2i) = 12 + 3i

Multiply and Divide

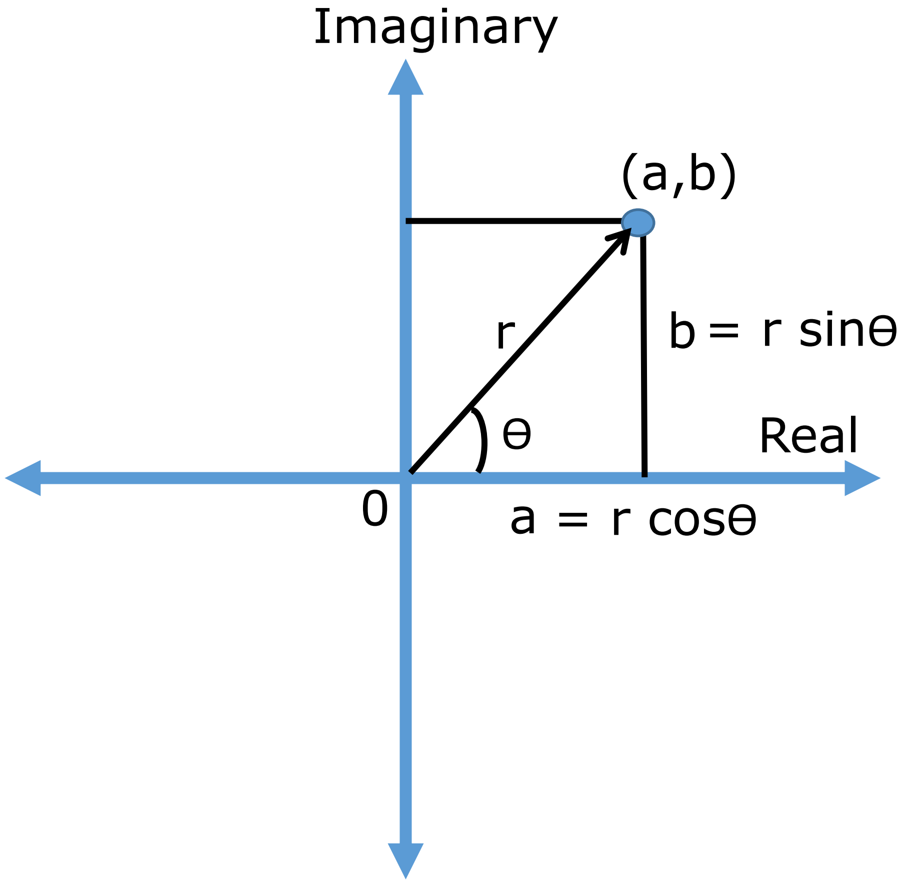
- Example 1:
  - c1 = 4 +5i, c2 = -8 +2i
- Multiply c1 and c2
  - Result = (4 +5i) x (-8 +2i) = -32 +8i -40i +10i^2 = -42 – 32i
- Divide c1 and c2

Division of Complex Numbers

- Complex conjugate
  - Changing the sign of the imaginary part only
  - c = a + bi
  - c* = a – bi (complex conjugate)
- Divide c1 / c2, c1 = 4 +5i, c2 = -8 +2i
  - Result = $\frac{(4+5i)\,(-8-2i)}{(-8+2i)\,(-8-2i)}$

    = -11/34 -12/17i

Polar representation



Complex plane / Argand plane

$r = \sqrt{a^2 + b^2}$
$a = r\cos\theta$
$b = r\sin\theta$

$c1 = a + bi$
$= r\cos\theta + (r\sin\theta)i$
$= r(\cos\theta + i\sin\theta)$
$= re^{i\theta}$

In a polar coordinate system, a point can be represented by a pair (r, ϴ), where r is the length of the straight line joining the point with the origin and ϴ is the angle with the x-axis.

(r, ϴ) – (Magnitude and Phase)

Complex Vector Space Fundamentals

Vectors in Quantum Computing

- The state of a quantum bit (qubit) is represented as vector
- A quantum system consists of a number of qubits, which change states as gate operations are carried out on them
  - We discuss about vector and vector spaces
- Vector space is collection of vectors with some defined properties

Vector Space

- Linear algebra is the study of vector spaces, and of linear operations on the vectors in those vector spaces.
- The basic objects of linear algebra are ***vector spaces***.
  - The vector space of most interest to us is $\mathbb{C}^n$, the space of all n-tuples of complex numbers, $(z_1, z_2, \ldots, z_n)$.
- The elements of a vector space are called ***vectors***, often represented as a column matrix:

$$\begin{bmatrix} z_1 \\ \vdots \\ z_n \end{bmatrix}$$

Operations on Vectors: Addition of Vectors

- There is an addition operation defined that operate on two vectors to produce a new vector. In $\mathbb{C}^n$, the addition operation for vectors is defined by

$$\begin{bmatrix} z_1 \\ \vdots \\ z_n \end{bmatrix} + \begin{bmatrix} z_1' \\ \vdots \\ z_n' \end{bmatrix} \equiv \begin{bmatrix} z_1 + z_1' \\ \vdots \\ z_n + z_n' \end{bmatrix}$$

where the addition operators on the right are ordinary additions of complex numbers.

An Example

- Consider two elements of $\mathbb{C}^4$ as: $V = \begin{bmatrix} 6-4i \\ 7+3i \\ 4.2-8.1i \\ -3i \end{bmatrix}$ and $W = \begin{bmatrix} 16+2.3i \\ -7i \\ 6 \\ -4i \end{bmatrix}$

- We can compute the sum vector $V + W \in \mathbb{C}^4$

$$\begin{bmatrix} 6-4i \\ 7+3i \\ 4.2-8.1i \\ -3i \end{bmatrix} + \begin{bmatrix} 16+2.3i \\ -7i \\ 6 \\ -4i \end{bmatrix} = \begin{bmatrix} (6-4i)+(16+2.3i) \\ (7+3i)+(-7i) \\ (4.2-8.1i)+(6) \\ (-3i)+(-4i) \end{bmatrix} = \begin{bmatrix} 22-1.7i \\ 7-4i \\ 10.2-8.1i \\ -7i \end{bmatrix}.$$

Vector Addition is Commutative and Associative

- Consider two elements of $\mathbb{C}^4$ as: V = $\begin{bmatrix} 6-4i \\ 7+3i \\ 4.2-8.1i \\ -3i \end{bmatrix}$ and W = $\begin{bmatrix} 16+2.3i \\ -7i \\ 6 \\ -4i \end{bmatrix}$
- V + W = W + V
- Consider three vectors V, W and X
  - (V + W) + X = V + (W + X)
- Vector addition holds commutative and associative properties

Special Vector: Zero Vector

- $0 = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$

- For all vectors $V \in \mathbb{C}^4$
  - $V + 0 = V = 0 + V$

Requirement of Additive Inverse

- Each vector also has an ***additive inverse***.

  Consider the vector in $\mathbb{C}^4$:

$$V = \begin{bmatrix} 6 - 4i \\ 7 + 3i \\ 4.2 - 8.1i \\ -3i \end{bmatrix}$$

Requirement of Additive Inverse

- There exists another vector –V in $\mathbb{C}^4$ such that:

$$-V = \begin{bmatrix} -6+4i \\ -7-3i \\ -4.2+8.1i \\ 3i \end{bmatrix} \in \mathbb{C}^4$$

$$V + (-V) = \begin{bmatrix} 6-4i \\ 7+3i \\ 4.2-8.1i \\ -3i \end{bmatrix} + \begin{bmatrix} -6+4i \\ -7-3i \\ -4.2+8.1i \\ 3i \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} = \mathbf{0}$$

The set $\mathbb{C}^4$ with the addition, inverse operation and zero such that the addition is associative and commutative form an Abelian Group.

Multiplication of a Scalar with a Vector

- Furthermore, in a vector space there is a "***multiplication by a scalar***" operation, which is defined in $\mathbb{C}^n$ as:

$$z \begin{bmatrix} z_1 \\ \vdots \\ z_n \end{bmatrix} \equiv \begin{bmatrix} z z_1 \\ \vdots \\ z z_n \end{bmatrix}$$

where z is a scalar (i.e. a complex number), and the multiplications on the right are simple complex number multiplications.

An Example

$$(3+2i)\cdot\begin{bmatrix}6+3i\\0+0i\\5+1i\\4\end{bmatrix}=\begin{bmatrix}12+21i\\0+0i\\13+13i\\12+8i\end{bmatrix}$$

Complex Vector Space

- $C^{m \times n}$, the set of all m-by-n matrices (two-dimensional arrays) with complex entries, is a complex vector space

$$
\begin{array}{cc}
 & \begin{array}{cccc} \mathbf{0} & \mathbf{1} & \cdots & \boldsymbol{n-1} \end{array} \\
A = \begin{array}{c} \mathbf{0} \\ \mathbf{1} \\ \vdots \\ \boldsymbol{m-1} \end{array} &
\begin{bmatrix}
c_{0,0} & c_{0,1} & \cdots & c_{0,n-1} \\
c_{1,0} & c_{1,1} & \cdots & c_{1,n-1} \\
\vdots & \vdots & \ddots & \vdots \\
c_{m-1,0} & c_{m-1,1} & \cdots & c_{m-1,n-1}
\end{bmatrix}
\end{array}
$$

Complex Vector Space cont.

- For C^{mxn} , when n = 1: $C^{mxn} = C^{mx1}$
  - **Hence vectors are special type of matrices**.

- Consider C^{mxn} , where m = n (Square matrix)
  - Here the vector space C^{nxn} has more operations and more structure than a complex vector space
  - The three operations are **Transpose, Complex Conjugate and Adjoint or Dagger**
  - These operations are important in the context of quantum gate operations.

Transpose of a Matrix

$$\mathbf{A} = \begin{bmatrix} 6-3i & 2+12i & -19i \\ 0 & 5+2.1i & 17 \\ 1 & 2+5i & 3-4.5i \end{bmatrix} \qquad \mathbf{A}^{\mathsf{T}} = \begin{bmatrix} 6-3i & 0 & 1 \\ 2+12i & 5+2.1i & 2+5i \\ -19i & 17 & 3-4.5i \end{bmatrix}$$

Complex Conjugate of a Matrix

$$\mathbf{A} = \begin{bmatrix} 6-3i & 2+12i & -19i \\ 0 & 5+2.1i & 17 \\ 1 & 2+5i & 3-4.5i \end{bmatrix} \qquad \mathbf{A^*} = \begin{bmatrix} 6+3i & 2-12i & 19i \\ 0 & 5-2.1i & 17 \\ 1 & 2-5i & 3+4.5i \end{bmatrix}$$

Adjoint/Dagger of a Matrix

$$\mathbf{A} = \begin{bmatrix} 6-3i & 2+12i & -19i \\ 0 & 5+2.1i & 17 \\ 1 & 2+5i & 3-4.5i \end{bmatrix}$$

$$\mathbf{A}^T = \begin{bmatrix} 6-3i & 0 & 1 \\ 2+12i & 5+2.1i & 2+5i \\ -19i & 17 & 3-4.5i \end{bmatrix}$$

$$(\mathbf{A}^T)^* = \begin{bmatrix} 6+3i & 0 & 1 \\ 2-12i & 5-2.1i & 2-5i \\ 19i & 17 & 3+4.5i \end{bmatrix}$$

$$\mathbf{A}^\dagger = \begin{bmatrix} 6+3i & 0 & 1 \\ 2-12i & 5-2.1i & 2-5i \\ 19i & 17 & 3+4.5i \end{bmatrix}$$

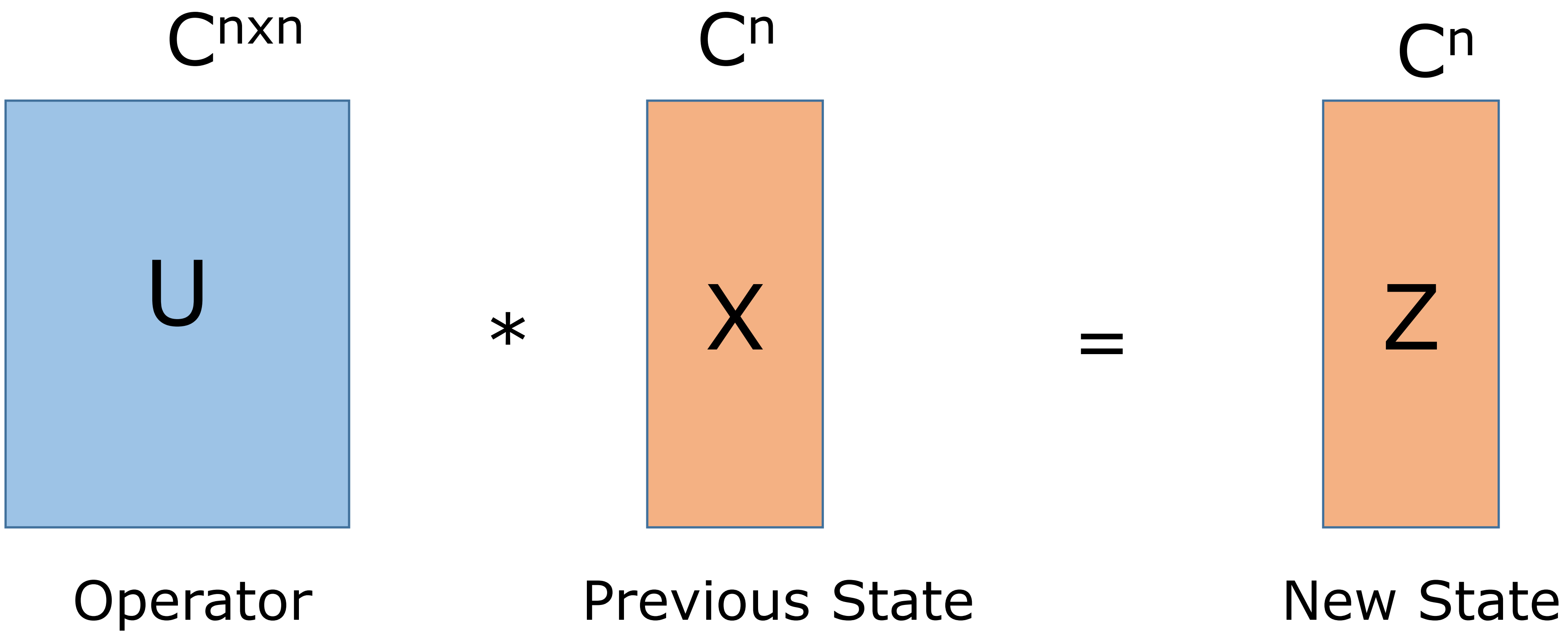
Some Operations on Matrix

- Every quantum gate operation can be represented as matrix (U)
- Every quantum state is represented by a vector (V)
- When V is applied as input to U, the new quantum state becomes U * V
- Every composite quantum gate operations (U1, U2) can be represented as a matrix obtained by multiplying the matrices corresponding to U1 and U2

Point to Remember

- Let U be any element in C^{nxn}:, for any element X $\in$ C^n, U * X is in C^n.
  - U: $C^n \rightarrow C^n$
- The elements of C^n represents the state of a quantum system
- Consider a state X $\in$ C^n, and a matrix U $\in$ C^{nxn}, if we perform U * X, then U * X is an element of C^n, which is nothing but another state of the system.

Example



Mathematically : $\hat{A}|\phi\rangle = |\phi'\rangle$ for this example:

$$\hat{U}|X\rangle = |Z\rangle$$

Summary

- Motivation of the course
- Quantum computing in general
- Complex numbers
- Complex Vector space
- Operation on matrices