January 15-19: Quantum Computing, Quantum Machine Learning and Quantum Information Theories

Morten Hjorth-Jensen[1,2]

Department of Physics and Center for Computing in Science Education, University of Oslo, Norway[1]

Department of Physics and Astronomy and Facility for Rare Isotope Beams, Michigan State University, East Lansing, Michigan, USA[2]

Jan 17, 2024

Overview of week first week, Basic Notions of Quantum Mechanics

Basics in Linear Algebra, The Hilbert Space, Operators on Hilbert Spaces, States and qubits

1. Mathematical notation, Hilbert spaces and operators
2. Description of Quantum Systems and one-qubit systems
3. States in Hilbert Space, pure and mixed states
4. Operators and simple gates

Reading recommendation: Scherer, Mathematics of Quantum Computations, chapter 2

Practicalities

1. Although the course is defined as a self-study course, we will have weekly lectures with small weekly exercise assignments
2. We plan to work on two projects which will define the content of the course, the format can be agreed upon by the participants but the following topics can be included
   - Quantum computing and simulation of quantum mechanical model systenms
   - Continuation of the first topic to more realistic systems or applications of quantum machine learning algorithms
3. Two project which count 50% each for the final grade
4. Deadline first project March 22
5. Deadline second project June 1
6. All info at the GitHub address https://github.com/CompPhysics/QuantumComputingMachineLearning

Notations and definitions

Throughout this course we will use the following notations.
Vectors, matrices and higher-order tensors are always boldfaced, with vectors given by lower case letter letters and matrices and higher-order tensors given by upper case letters.
Unless otherwise stated, the elements v_i of a vector $\boldsymbol{v}$ are assumed to be real. That is a vector of length n is defined as $\boldsymbol{x} \in \mathbb{R}^n$ and if we have a complex vector we have $\boldsymbol{x} \in \mathbb{C}^n$.
For a matrix of dimension $n \times n$ we have $\boldsymbol{A} \in \mathbb{R}^{n\times n}$ and the first matrix element starts with row element (row-wise ordering) zero and column element zero.

Some mathematical notations

1. For all/any $\forall$
2. Implies $\implies$
3. Equivalent $\equiv$
4. Real variable $\mathbb{R}$
5. Integer variable $\mathbb{I}$
6. Complex variable $\mathbb{C}$

Vectors

We start by defining a vector $\boldsymbol{x}$ with n components, with x_0 as our first element, as

$$\boldsymbol{x} = \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ \dots \\ \dots \\ x_{n-1} \end{bmatrix}.$$

and its transpose

$$\boldsymbol{x}^T = \begin{bmatrix} x_0 & x_1 & x_2 & \dots & \dots & x_{n-1} \end{bmatrix},$$

In case we have a complex vector we define the hermitian conjugate

$$\boldsymbol{x}^\dagger = \begin{bmatrix} x_0^* & x_1^* & x_2^* & \dots & \dots & x_{n-1}^* \end{bmatrix},$$

With a given vector $\boldsymbol{x}$, we define the inner product as

$$\boldsymbol{x}^T\boldsymbol{x} = \sum_{i=0}^{n-1} x_i x_i = x_0^2 + x_1^2 + \cdots + x_{n-1}^2.$$

Outer products

In addition to inner products between vectors/states, the outer product plays a central role in many applications. It is defined as

$$
\boldsymbol{x}\boldsymbol{y}^T = \begin{bmatrix}
x_0 y_0 & x_0 y_1 & x_0 y_2 & \dots & \dots & x_0 y_{n-2} & x_0 y_{n-1} \\
x_1 y_0 & x_1 y_1 & x_1 y_2 & \dots & \dots & x_1 y_{n-2} & x_1 y_{n-1} \\
x_2 y_0 & x_2 y_1 & x_2 y_2 & \dots & \dots & x_2 y_{n-2} & x_2 y_{n-1} \\
\dots & \dots & \dots & \dots & \dots & \dots & \dots \\
\dots & \dots & \dots & \dots & \dots & \dots & \dots \\
x_{n-2} y_0 & x_{n-2} y_1 & x_{n-2} y_2 & \dots & \dots & x_{n-2} y_{n-2} & x_{n-2} y_{n-1} \\
x_{n-1} y_0 & x_{n-1} y_1 & x_{n-1} y_2 & \dots & \dots & x_{n-1} y_{n-2} & x_{n-1} y_{n-1}
\end{bmatrix}
$$

The latter defines also our basic matrix layout.

Basic Matrix Features

A general $n \times n$ matrix is given by

$$\boldsymbol{A} = \begin{bmatrix} a_{00} & a_{01} & a_{02} & \dots & \dots & a_{0n-2} & a_{0n-1} \\ a_{10} & a_{11} & a_{12} & \dots & \dots & a_{1n-2} & a_{1n-1} \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ a_{n-20} & a_{n-21} & a_{n-22} & \dots & \dots & a_{n-2n-2} & a_{n-2n-1} \\ a_{n-10} & a_{n-11} & a_{n-12} & \dots & \dots & a_{n-1n-2} & a_{n-1n-1} \end{bmatrix},$$

or in terms of its column vectors $\boldsymbol{a}_i$ as

$$\boldsymbol{A} = \begin{bmatrix} \boldsymbol{a}_0 & \boldsymbol{a}_1 & \boldsymbol{a}_2 & \dots & \dots & \boldsymbol{a}_{n-2} & \boldsymbol{a}_{n-1} \end{bmatrix}.$$

We can think of a matrix as a diagram of in general n rowns and m columns. In the example here we have a square matrix.

The inverse of a matrix

The inverse of a square matrix (if it exists) is defined by

$$\boldsymbol{A}^{-1} \cdot \boldsymbol{A} = \boldsymbol{I},$$

where $\boldsymbol{I}$ is the unit matrix.

Basic Matrix Features

Matrix Properties Reminder

Relations	Name	matrix elements
$A = A^T$	symmetric	$a_{ij} = a_{ji}$
$A = \left(A^T\right)^{-1}$	real orthogonal	$\sum_k a_{ik} a_{jk} = \sum_k a_{ki} a_{kj} = \delta_{ij}$
$A = A^*$	real matrix	$a_{ij} = a_{ij}^*$
$A = A^\dagger$	hermitian	$a_{ij} = a_{ji}^*$
$A = \left(A^\dagger\right)^{-1}$	unitary	$\sum_k a_{ik} a_{jk}^* = \sum_k a_{ki}^* a_{kj} = \delta_{ij}$

Some famous Matrices

- Diagonal if $a_{ij} = 0$ for $i \neq j$
- Upper triangular if $a_{ij} = 0$ for $i > j$
- Lower triangular if $a_{ij} = 0$ for $i < j$
- Upper Hessenberg if $a_{ij} = 0$ for $i > j + 1$
- Lower Hessenberg if $a_{ij} = 0$ for $i < j + 1$
- Tridiagonal if $a_{ij} = 0$ for $|i - j| > 1$
- Lower banded with bandwidth p: $a_{ij} = 0$ for $i > j + p$
- Upper banded with bandwidth p: $a_{ij} = 0$ for $i < j + p$
- Banded, block upper triangular, block lower triangular....

Matrix Features

Some Equivalent Statements

For an $n \times n$ matrix $\boldsymbol{A}$ the following properties are all equivalent

- If the inverse of $\boldsymbol{A}$ exists, $\boldsymbol{A}$ is nonsingular.
- The equation $\boldsymbol{Ax} = 0$ implies $\boldsymbol{x} = 0$.
- The rows of $\boldsymbol{A}$ form a basis of R^N.
- The columns of $\boldsymbol{A}$ form a basis of R^N.
- $\boldsymbol{A}$ is a product of elementary matrices.
- 0 is not an eigenvalue of $\boldsymbol{A}$.

Important Mathematical Operations

The basic matrix operations that we will deal with are addition and subtraction

$$\boldsymbol{A} = \boldsymbol{B} \pm \boldsymbol{C} \Longrightarrow a_{ij} = b_{ij} \pm c_{ij},$$

and scalar-matrix multiplication

$$\boldsymbol{A} = \gamma \boldsymbol{B} \Longrightarrow a_{ij} = \gamma b_{ij}.$$

Vector-matrix and Matrix-matrix multiplication

We have also vector-matrix multiplications

$$\boldsymbol{y} = \boldsymbol{A}\boldsymbol{x} \Longrightarrow y_i = \sum_{j=1}^{n} a_{ij} x_j,$$

and matrix-matrix multiplications

$$\boldsymbol{A} = \boldsymbol{B}\boldsymbol{C} \Longrightarrow a_{ij} = \sum_{k=1}^{n} b_{ik} c_{kj},$$

and transpositions of a matrix

$$\boldsymbol{A} = \boldsymbol{B}^T \Longrightarrow a_{ij} = b_{ji}.$$

Important Mathematical Operations

Similarly, important vector operations that we will deal with are addition and subtraction

$$\boldsymbol{x} = \boldsymbol{y} \pm \boldsymbol{z} \Longrightarrow x_i = y_i \pm z_i,$$

scalar-vector multiplication

$$\boldsymbol{x} = \gamma \boldsymbol{y} \Longrightarrow x_i = \gamma y_i,$$

Other important mathematical operations

and vector-vector multiplication (called Hadamard multiplication)

$$\boldsymbol{x} = \boldsymbol{y}\boldsymbol{z} \Longrightarrow x_i = y_i z_i.$$

Finally, as already metnioned, the inner or so-called dot product resulting in a constant

$$x = \boldsymbol{y}^T \boldsymbol{z} \Longrightarrow x = \sum_{j=1}^{n} y_j z_j,$$

and the outer product, which yields a matrix,

$$\boldsymbol{A} = \boldsymbol{y}\boldsymbol{z}^T \Longrightarrow a_{ij} = y_i z_j,$$

Defining basis states and quantum mechanical operators

We extend now to quantum mechanics our definitions of vectors, matrices and more.

We start by defining a state vector $\boldsymbol{x}$ (meant to represent various quantum mechanical degrees of freedom) with n components as

$$\boldsymbol{x} = \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ \dots \\ \dots \\ x_{n-1} \end{bmatrix}.$$

Throughout these notes we will use the so-called Dirac **bra-ket** formalism and we will replace the above standard boldfaced notation for a vector with

$$\boldsymbol{x} = |x\rangle = \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ \dots \end{bmatrix},$$

Inner product in Dirac notation

With a given vector $|x\rangle$, we define the inner product as

$$\langle x|x\rangle = \sum_{i=0}^{n-1} x_i^* x_i = x_0^2 + x_1^2 + \cdots + x_{n-1}^2.$$

For two arbitrary vectors $|x\rangle$ and $|y\rangle$ with the same lentgh, we have the general expression

$$\langle y|x\rangle = \sum_{i=0}^{n-1} y_i^* x_i = y_0^* x_0 + y_1^* x_1 + \cdots + y_{n-1}^* x_{n-1}.$$

The inner product is a real number

Note well that the inner product $\langle x|x\rangle$ is always a real number while for a two different vectors $\langle y|x\rangle$ is in general not equal to $\langle x|y\rangle$, as can be seen from the following example
We note in bypassing that $|x\rangle^\dagger = \langle x|$, $\langle x|^\dagger = |x\rangle$ and $(|x\rangle^\dagger)^\dagger = |x\rangle$.

Examples

Let us assume that $|x\rangle$ is given by

$$|x\rangle = \begin{bmatrix} 1-\imath \\ 2+\imath \end{bmatrix}.$$

The inner product gives us

$$\langle x|x\rangle = (1+\imath)(1-\imath) + (2-\imath)(2+\imath) = 7,$$

a real number.

Norm

We can use the norm/inner product to normalize the vector $|x\rangle$ and obtain

$$|x\rangle = \frac{1}{\sqrt{7}} \begin{bmatrix} 1 - \imath \\ 2 + \imath \end{bmatrix}.$$

As another example, consider the two vectors

$$|x\rangle = \begin{bmatrix} -1 \\ 2\imath \\ 1 \end{bmatrix},$$

and

$$|y\rangle = \begin{bmatrix} 1 \\ 0\imath \\ \imath \end{bmatrix}.$$

We see that the inner products $\langle x|y\rangle = -1 + \imath$, which is not the same as $\langle y|x\rangle = -1 - \imath$. This leads to the important rule

$$\langle x|y\rangle^* = \langle y|x\rangle.$$

Outer products

In addition to inner products between vectors/states, the outer product plays a central role in all of quantum mechanics. It is defined as

$$|x\rangle\langle y| = \begin{bmatrix} x_0 y_0^* & x_0 y_1^* & x_0 y_2^* & \dots & \dots & x_0 y_{n-2}^* & x_0 y_{n-1}^* \\ x_1 y_0^* & x_1 y_1^* & x_1 y_2^* & \dots & \dots & x_1 y_{n-2}^* & x_1 y_{n-1}^* \\ x_2 y_0^* & x_2 y_1^* & x_2 y_2^* & \dots & \dots & x_2 y_{n-2}^* & x_2 y_{n-1}^* \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ x_{n-2} y_0^* & x_{n-2} y_1^* & x_{n-2} y_2^* & \dots & \dots & x_{n-2} y_{n-2}^* & x_{n-2} y_{n-1}^* \\ x_{n-1} y_0^* & x_{n-1} y_1^* & x_{n-1} y_2^* & \dots & \dots & x_{n-1} y_{n-2}^* & x_{n-1} y_{n-1}^* \end{bmatrix}$$

Different operators and gates

In quantum computing, the so-called Pauli matrices, and other simple 2×2 matrices, play an important role, ranging from the setup of quantum gates to a rewrite of creation and annihilation operators and other quantum mechanical operators. Let us start with the familiar Pauli matrices and remind ourselves of some of their basic properties.

The Pauli matrices are defined as

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

$$\sigma_y = \begin{bmatrix} 0 & -\imath \\ \imath & 0 \end{bmatrix},$$

and

$$\sigma_z = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}.$$

Properties of Pauli matrices

It is easy to show that the matrices obey the properties (being involutory)

$$\sigma_x\sigma_x = \sigma_y\sigma_y = \sigma_z\sigma_z = \boldsymbol{I} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix},$$

that is their products with themselves result in the identity matrix $\boldsymbol{I}$. Furthermore, the Pauli matrices are unitary matrices meaning that their inverses are equal to their hermitian conjugated matrices. The determinants of the Pauli matrices are all equal to -1, as can be easily verified.

The Pauli matrices obey also the following commutation rules

$$[\sigma_x, \sigma_y] = 2\imath\sigma_z.$$

Before we proceed with other matrices and how they can be used to operate on various quantum mechanical states, let us try to define various basis sets and their pertinent notations. We will often refer to these basis states as our computational basis.

Definition of Computational basis states

Assume we have a two-level system where the two states are represented by the state vectors $|\phi_0\rangle$ and $|\phi_1\rangle$, respectively. These states could represent selected or effective degrees of freedom for either a single particle (fermion or boson) or they could represent effective many-body degrees of freedon. In actual realizations of quantum computing we search often for candidate systems where we can use some low-lying states as computational basis states. But we are not limited to quantum computing. When doing many-body physics, due to the exploding degrees of freedom, we normally search after effective ways by which we can reduce the involved dimensionalities to a number of degrees of freedom we can handle by a given many-body method.

Projection operators

We will now relabel the above two states as two orthogonal and normalized basis (ONB) states

$$|\phi_0\rangle = |0\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix},$$

and

$$|\phi_1\rangle = |1\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix}.$$

Identity and projection operators

It is straight forward to see that $\langle 1|0\rangle = 0$. With these two states we can define the define the identity operator $\boldsymbol{I}$ as the sum of the outer products of these two states, namely

$$\boldsymbol{I} = \sum_{i=0}^{i=1} |i\rangle\langle i| = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}.$$

We can further define the projection operators

$$\boldsymbol{P} = |0\rangle\langle 0| = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix},$$

and

$$\boldsymbol{Q} = |1\rangle\langle 1| = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}.$$

We note that $P^2 = P$, $Q^2 = Q$ (the operators are idempotent) and that their determinants are zero, meaning in turn that we cannot use these operators for unitary/orthogonal transformations. However, they play important roles in defining effective Hilbert spaces for many-body studies. Finally, before proceeding we note also that the two matrices commute and we have $\boldsymbol{PQ} = 0$ and

Superposition and more

Using the properties of ONBs we can expand a new state in terms of the above states. These states could also form a basis which is an eigenbasis of a selected Hamiltonian (more of this below).
We define now a new state which is a linear expansion in terms of these computational basis states

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

where the coefficients $\alpha = \langle 0|\psi\rangle$ and $\beta = \langle 1|\psi\rangle$ reresent the overlaps between the computational basis states and the state $|\psi\rangle$.
In quantum speech, we say the state is in a superposition of the states $|0\rangle$ and $|1\rangle$.

Inner products

Computing the inner product of $|\psi\rangle$ we obtain

$$\langle\psi|\psi\rangle = |\alpha|^2\langle 0|0\rangle + |\beta|^2\langle 1|1\rangle = |\alpha|^2 + |\beta|^2 = 1,$$

since the new basis, which is defined in terms of a a unitary/orthogonal transformation, preserves the orthogonality and norm of the original computational basis $|0\rangle$ and $|1\rangle$. To see this, consider the unitary transformation (show derivation of preserving orthogonality).

Acting with projection operators

If we now act with the projection operators $\boldsymbol{P}$ and $\boldsymbol{Q}$ on the state $|\psi\rangle$ we get

$$\boldsymbol{P}|\psi\rangle = |0\rangle\langle 0|(\alpha|0\rangle + \beta|1\rangle) = \alpha|0\rangle,$$

that is we **project** out the $|0\rangle$ component of the state $|\psi\rangle$ with the coefficient α while $\boldsymbol{Q}$ projects out the $|1\rangle$ component with coefficient β as seen from

$$\boldsymbol{Q}|\psi\rangle = |1\rangle\langle 1|(\alpha|0\rangle + \beta|1\rangle) = \beta|1\rangle.$$

The above results can easily be derived by multiplying the pertinent matrices with the vectors $|0\rangle$ and $|1\rangle$, respectively.

Density matrix

Using the above linear expansion we can now define the density matrix of the state $|\psi\rangle$ as the outer product

$$\boldsymbol{\rho} = |\psi\rangle\langle\psi| = \alpha\alpha^*|0\rangle\langle 0|+\alpha\beta^*|0\rangle\langle 1|+\beta\alpha^*|1\rangle\langle 0|+\beta\beta^*|1\rangle\langle 1| = \begin{bmatrix}\alpha\alpha^* & \alpha\beta^* \\ \beta\alpha^* & \beta\beta^*\end{bmatrix}$$

Finally, we note that the trace of the density matrix is simply given by unity

$$\mathrm{tr}\boldsymbol{\rho} = \alpha\alpha^* + \beta\beta^* = 1.$$

Other important matrices

We present other operators (as matrices) which play an important role in quantum computing, the so-called Hadamard matrix (or gate as we will use later)

$$\boldsymbol{H} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}.$$

The action of the operator $\boldsymbol{H}$ on a computational basis state like $|0\rangle$ gives

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

and

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

that is we create a superposition of the states $|0\rangle$ and $|1\rangle$.

Phase matrix

Another famous operation is the phase matrix given by

$$\boldsymbol{S} = \begin{bmatrix} 1 & 0 \\ 0 & \imath \end{bmatrix}.$$

Tensor products

Consider now two vectors with length $n = 2$, with elements

$$|x\rangle = \begin{bmatrix} x_0 \\ x_1 \end{bmatrix},$$

and

$$|y\rangle = \begin{bmatrix} y_0 \\ y_1 \end{bmatrix}.$$

The tensor product of these two vectors is defined as

$$|x\rangle \otimes |y\rangle = |xy\rangle = \begin{bmatrix} x_0 y_0 \\ x_0 y_1 \\ x_1 y_0 \\ x_1 y_1 \end{bmatrix},$$

which is now a vector of length 4.

Examples of tensor products

If we now go back to our original one-qubit basis states, we can form teh following tensor products

$$|0\rangle \otimes |0\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \otimes \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} = |00\rangle,$$

$$|0\rangle \otimes |1\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \otimes \begin{bmatrix} 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} = |01\rangle.$$

More states

$$|1\rangle \otimes |0\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \otimes \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix} = |10\rangle,$$

and finally

$$|1\rangle \otimes |1\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \otimes \begin{bmatrix} 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} = |11\rangle.$$

Three qubits

We have now four different states and we could try to make a new list by relabeling the states as follows $|00\rangle = |0\rangle$, $|01\rangle = |1\rangle$, $|10\rangle = |2\rangle$, $|11\rangle = |3\rangle$.

In similar ways we can define the tensor product of three qubits (or single-particle states) as

$$|0\rangle \otimes |0\rangle \otimes |0\rangle = \begin{bmatrix}1\\0\end{bmatrix} \otimes \begin{bmatrix}1\\0\end{bmatrix} \otimes \begin{bmatrix}1\\0\end{bmatrix} = \begin{bmatrix}1\\0\\0\\0\\0\\0\\0\\0\end{bmatrix} = |000\rangle,$$

which is a new vector of length eight. We note that with a single-particle basis given the states $|0\rangle$ and $|1\rangle$ we can, with N particles construct 2^N different states. This is something we can generalize to

- discuss ways of labeling states

Tensor products of matrices

The tensor product of two 2×2 matrices $\boldsymbol{A}$ and $\boldsymbol{B}$ is given by

$$\boldsymbol{A} \times \boldsymbol{B} = \begin{bmatrix} a_{00} & a_{01} \\ a_{10} & a_{11} \end{bmatrix} \otimes \begin{bmatrix} b_{00} & b_{01} \\ b_{10} & b_{11} \end{bmatrix} = \begin{bmatrix} a_{00}b_{00} & a_{00}b_{01} & a_{01}b_{00} & a_{01}b_{01} \\ a_{00}b_{10} & a_{00}b_{11} & a_{01}b_{10} & a_{01}b_{11} \\ a_{10}b_{00} & a_{10}b_{01} & a_{11}b_{00} & a_{11}b_{01} \\ a_{10}b_{10} & a_{10}b_{11} & a_{11}b_{10} & a_{11}b_{11} \end{bmatrix}$$

Measurements

The probability of a measurement on a quantum system giving a certain result is determined by the weight of the relevant basis state in the state vector. After the measurement, the system is in a state that corresponds to the result of the measurement. The operators and gates discussed below are examples of operations we can perform on specific states.
We consider the state

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

Definitions of measurements

1. A measurement can yield only one of the above states, either $|0\rangle$ or $|1\rangle$.
2. The probability of a measurement resulting in $|0\rangle$ is $\alpha^*\alpha = |\alpha|^2$.
3. The probability of a measurement resulting in $|1\rangle$ is $\beta^*\beta = |\beta|^2$.
4. And we note that the sum of the outcomes gives $\alpha^*\alpha + \beta^*\beta = 1$ since the two states are normalized.

After the measurement, the state of the system is the state associated with the result of the measurement.
We have already encountered the projection operators P and Q.
Let us now look at other types of operations we can make on qubit states.

Different operators and gates

In quantum computing, the so-called Pauli matrices, and other simple 2×2 matrices, play an important role, ranging from the setup of quantum gates to a rewrite of creation and annihilation operators and other quantum mechanical operators. Let us start with the familiar Pauli matrices and remind ourselves of some of their basic properties.

Assume we operate with σ_x on our basis state $|0\rangle$. This gives

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

that is we switch from $|0\rangle$ to $|1\rangle$ (often called a spin flip operation) and similary we have

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

More on Pauli matrices

This matrix plays an important role in quantum computing since we can link this with the classical **NOT** operation. If we send in bit 0, the **NOT** gate outputs bit 1 and vice versa. We can use the σ_x matrix to implement the quantum mechanical equivalent of a classical **NOT** gate. If we input what we could represent as bit 0 in terms of the basis state $|0\rangle$, operating on this state results in the state $|1\rangle$, which we in turn can interpret as the classical bit 1.

Linear superposition

If we have a linear superposition of these states we obtain

$$\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} \alpha \\ \beta \end{bmatrix} = \begin{bmatrix} \beta \\ \alpha \end{bmatrix}.$$

The σ_y matrix introduces an imaginary sign, which we will later encounter in terms of so-called phase-shift operations.

The σ_z matrix

The σ_z matrix has the following effect

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

and

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

which we will also link with a specific phase-shift.

Unitarity

The matrices we introduced here are so-called unitary matrices. This is an important element in quantum mechanics since the evolution of a closed quantum system is described by operations involving unitary operations only.
We have defined a new state $|\psi_p\rangle$ as a linear expansion in terms of an orthogonal and normalized basis (our computational basis) ϕ_λ

$$|\psi_i\rangle = \sum_j u_{ij}|\phi_j\rangle. \tag{1}$$

Hamiltonians

It is normal to choose a basis defined as the eigenfunctions of parts of the full Hamiltonian. The typical situation consists of the solutions of the one-body part of the Hamiltonian, that is we have

$$\hat{h}_0|\phi_i\rangle = \epsilon_i|\phi_i\rangle.$$

This is normally referred to as a single-particle basis $|\phi_i(\mathrm{r})\rangle$, defined by the quantum numbers i and r.

Unitary transformations

A unitary transformation is important since it keeps the orthogonality. To see this consider first a basis of vectors v_i,

$$\mathrm{v}_i = \begin{bmatrix} v_{i1} \\ \dots \\ \dots \\ v_{in} \end{bmatrix}$$

We assume that the basis is orthogonal, that is

$$\mathrm{v}_j^T \mathrm{v}_i = \delta_{ij}.$$

An orthogonal or unitary transformation

$$\mathrm{w}_i = \mathrm{U}\mathrm{v}_i,$$

preserves the dot product and orthogonality since

$$\mathrm{w}_j^T \mathrm{w}_i = (\mathrm{U}\mathrm{v}_j)^T \mathrm{U}\mathrm{v}_i = \mathrm{v}_j^T \mathrm{U}^T \mathrm{U}\mathrm{v}_i = \mathrm{v}_j^T \mathrm{v}_i = \delta_{ij}.$$

Orthogonality preserved

This means that if the coefficients $u_{p\lambda}$ belong to a unitary or orthogonal transformation (using the Dirac bra-ket notation)

$$|\psi_i\rangle = \sum_j u_{ij}|\phi_j\rangle.$$

orthogonality is preserved.
This propertry is extremely useful when we build up a basis of many-body determinant based states.
Note also that although a basis $\{|\phi_i\rangle\}$ contains an infinity of states, for practical calculations we have always to make some truncations.

Example

Assume we have two one-qubit states represented by

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

and

$$|\phi\rangle = \gamma|0\rangle + \delta|1\rangle = \begin{bmatrix} \gamma \\ \delta \end{bmatrix}.$$

We assume that the state $|\phi\rangle$ is obtained through a unitary transformation of $|\psi\rangle$ through a matrix $\boldsymbol{U}$ with its hermitian conjugate $\boldsymbol{U}^\dagger$ with matrix elements $u_{ij}^\dagger = u_{ji}^*$ and $\boldsymbol{I} = \boldsymbol{U}\boldsymbol{U}^\dagger = \boldsymbol{U}^\dagger\boldsymbol{U}$.

Inverse of unitary matrices

Note that this means that the hermitian conjugate of a unitary matrix is equal to its inverse. This has important consequences for what is called reversibility. We say quantum mechanics is a theory which is reversible with a probabilistic determinism. Classical mechanics on the other is reversible in a deterministic way, that is, knowing all initial conditions we can in principle determine the future motion of an object which obey the laws of motion of classical mechanics.
We have then

$$\begin{bmatrix}\gamma \\ \delta\end{bmatrix} = \begin{bmatrix}u_{00} & u_{01} \\ u_{10} & u_{11}\end{bmatrix}\begin{bmatrix}\alpha \\ \beta\end{bmatrix}.$$

New basis is also orthogonal

Since our original basis $|\psi\rangle$ is orthogonal and normalized with $|\alpha|^2 + |\beta|^2 = 1$, the new basis is also orthogonal and normalized, as we can see below here.

Since the inverse of a hermitian matrix is equal to its hermitian conjugate/adjoint), unitary transformations are always reversible.

Why are only unitary transformations allowed? The key lies in the way the inner product tranforms.

To see this we rewrite the new basis from the previous example in its two components as

$$|\phi\rangle_i = \sum_j u_{ij} |\psi\rangle_j,$$

or in terms of a matrix-vector notatio we have

$$|\phi\rangle = \boldsymbol{U} |\psi\rangle,$$

More on orthogonality

We have already assumed that $\langle\psi|\psi\rangle = |\alpha|^2 + |\beta|^2 = 1$.
We have that

$$\langle\phi|_i = \sum_j u_{ij}^* \langle\psi|_j,$$

or in terms of a matrix-vector notation we have

$$\langle\phi| = \langle\psi|\boldsymbol{U}^\dagger.$$

Note that the two vectors are row vectors now.
If we stay with this notation we have

$$\langle\phi|\phi\rangle = \langle\psi\boldsymbol{U}^\dagger\boldsymbol{U}|\psi\rangle = \langle\psi|\psi\rangle = 1!$$

Unitary transformations are rotations in state space which preserve the length (the square root of the inner product) of the state vector.