Appendix A
Linear Algebra for Quantum Computation

The goal of this appendix is to compile the definitions, notations, and facts of linear algebra that are important for this book. Quantum computation has inherited linear algebra from quantum mechanics as the supporting language for describing this area. It is essential to have a solid knowledge of the basic results of linear algebra to understand quantum computation and quantum algorithms. If the reader does not have this base knowledge, we suggest reading some basic references recommended at the end of this appendix.

A.1 Vector Spaces

A *vector space* V over the field of complex numbers $\mathbb{C}$ is a nonempty set of elements called vectors together with two operations called vector addition and multiplication of a vector by a scalar in $\mathbb{C}$. The addition operation is associative and commutative and satisfies the following axioms:

- There is an element $\mathbf{0} \in V$, such that, for each $\mathbf{v} \in V$, $\mathbf{v} + \mathbf{0} = \mathbf{0} + \mathbf{v} = \mathbf{v}$ (existence of neutral element).
- For each $\mathbf{v} \in V$, there exists $\mathbf{u} = (-1)\mathbf{v}$ in V such that $\mathbf{v} + \mathbf{u} = \mathbf{0}$ (existence of inverse element).

$\mathbf{0}$ is called zero vector. The scalar multiplication operation satisfies the following axioms:

- $a.(b.\mathbf{v}) = (a.b).\mathbf{v}$ (associativity),
- $1.\mathbf{v} = \mathbf{v}$ (1 is the neutral element of multiplication),
- $(a + b).\mathbf{v} = a.\mathbf{v} + b.\mathbf{v}$ (distributivity over sum of scalars),
- $a.(\mathbf{v} + \mathbf{w}) = a.\mathbf{v} + a.\mathbf{w}$ (distributivity over vector addition).

where $\mathbf{v}, \mathbf{w} \in V$ and $a, b \in \mathbb{C}$.

A vector space can be infinite, but in most applications in *quantum computation*, *finite vector spaces* are used and are denoted by $\mathbb{C}^n$, where n is the number of

R. Portugal, *Quantum Walks and Search Algorithms*, Quantum Science and Technology, https://doi.org/10.1007/978-3-319-97813-0

dimensions. In this case, the vectors have n complex entries. In this book, we rarely use infinite spaces, and in these few cases, we are interested only in finite subspaces. In the context of *quantum mechanics*, *infinite vector spaces* are used more frequently than finite spaces.

A *basis* for $\mathbb{C}^n$ consists of exactly n linearly independent vectors. If $\{\mathbf{v}_1, \ldots, \mathbf{v}_n\}$ is a basis for $\mathbb{C}^n$, then an arbitrary vector $\mathbf{v}$ can be written as

$$\mathbf{v} = \sum_{i=1}^{n} a_i \mathbf{v}_i,$$

where coefficients a_i are complex numbers. The *dimension* of a vector space is the number of basis vectors and is denoted by $\dim(V)$.

A.2 Inner Product

The *inner product* is a binary operation $(\cdot, \cdot) : V \times V \mapsto \mathbb{C}$, which obeys the following properties:

1. $(\cdot, \cdot)$ is linear in the second argument

$$\left(\mathbf{v}, \sum_{i=1}^{n} a_i \mathbf{v}_i\right) = \sum_{i=1}^{n} a_i \left(\mathbf{v}, \mathbf{v}_i\right).$$

2. $(\mathbf{v}_1, \mathbf{v}_2) = (\mathbf{v}_2, \mathbf{v}_1)^*$.
3. $(\mathbf{v}, \mathbf{v}) \geq 0$. The equality holds if and only if $\mathbf{v} = \mathbf{0}$.

In general, the inner product is not linear in the first argument. The property in question is called *conjugate-linear*.

There is more than one way to define an inner product on a vector space. In $\mathbb{C}^n$, the most used inner product is defined as follows: If

$$\mathbf{v} = \begin{bmatrix} a_1 \\ \vdots \\ a_n \end{bmatrix}, \quad \mathbf{w} = \begin{bmatrix} b_1 \\ \vdots \\ b_n \end{bmatrix},$$

then

$$(\mathbf{v}, \mathbf{w}) = \sum_{i=1}^{n} a_i^* b_i.$$

This expression is equivalent to the matrix product of the transpose–conjugate vector $\mathbf{v}^\dagger$ and $\mathbf{w}$.

Two vectors $\mathbf{v}_1$ and $\mathbf{v}_2$ are *orthogonal* if the inner product $(\mathbf{v}_1, \mathbf{v}_2)$ is zero. We also introduce the notion of *norm* using the inner product. The norm of $\mathbf{v}$, denoted by $\|\mathbf{v}\|$, is defined as

$$\|\mathbf{v}\| = \sqrt{(\mathbf{v}, \mathbf{v})}.$$

A *normalized vector* or *unit vector* is a vector whose norm is equal to 1. A basis is said *orthonormal* if all vectors are normalized and mutually orthogonal.

A finite vector space with an inner product is called a *Hilbert space* and denoted by $\mathcal{H}$. In order to an infinite vector space be a Hilbert space, it must obey additional properties besides having an inner product. Since we deal primarily with finite vector spaces, we use the term *Hilbert space* as a synonym for *vector space with an inner product*. A *vector subspace* (or simply subspace) W of a finite Hilbert space V is also a Hilbert space. The set of vectors orthogonal to all vectors of W is the Hilbert space $W^\perp$ called *orthogonal complement*. V is the direct sum of W and $W^\perp$, that is, $V = W \oplus W^\perp$. A N-dimensional Hilbert space is denoted by $\mathcal{H}^N$ to highlight its dimension. A Hilbert space associated with a system A is denoted by $\mathcal{H}_A$ or simply $\mathcal{A}$. If $\mathcal{A}$ is a subspace of $\mathcal{H}$, then $\mathcal{H} = \mathcal{A} + \mathcal{A}^\perp$, which means that any vector in $\mathcal{H}$ can be written as a sum of a vector in $\mathcal{A}$ and a vector in $\mathcal{A}^\perp$.

Exercise A.1. Let $\mathcal{A}$ and $\mathcal{B}$ be subspaces of $\mathcal{H}$. Show that $\dim(\mathcal{A} + \mathcal{B}) = \dim(\mathcal{A}) + \dim(\mathcal{B}) - \dim(\mathcal{A} \cap \mathcal{B})$, $\big(\mathcal{A} + \mathcal{B}\big)^\perp = \mathcal{A}^\perp \cap \mathcal{B}^\perp$, and $\big(\mathcal{A} \cap \mathcal{B}\big)^\perp = \mathcal{A}^\perp + \mathcal{B}^\perp$.

Exercise A.2. Give one example of subspaces $\mathcal{A}$ and $\mathcal{B}$ of $\mathbb{C}^3$ such that $(\mathcal{A} \cap \mathcal{B})^\perp \neq \mathcal{A} \cap \mathcal{B}^\perp + \mathcal{A}^\perp \cap \mathcal{B} + \mathcal{A}^\perp \cap \mathcal{B}^\perp$.

A.3 The Dirac Notation

In this review of linear algebra, we use the *Dirac* or *bra–ket notation*, which was introduced by the English physicist Paul Dirac in the context of quantum mechanics to aid algebraic manipulations. This notation is very easy to grasp. Several alternative notations for vectors are used, such as $\mathbf{v}$ and $\vec{v}$. The Dirac notation uses $|v\rangle$. Up to this point, instead of using boldface or an arrow over letter v, we put letter v between a vertical bar and a right angle bracket. If we have an indexed basis, that is, $\{\mathbf{v}_1, \ldots, \mathbf{v}_n\}$, in the Dirac notation we use the form $\{|v_1\rangle, \ldots, |v_n\rangle\}$ or $\{|1\rangle, \ldots, |n\rangle\}$. Note that if we are using a single basis, letter $\mathbf{v}$ is unnecessary in principle. Computer scientists usually start counting from 0. So, the first basis vector is usually called $\mathbf{v}_0$. In the Dirac notation we have

$$\mathbf{v}_0 = |0\rangle.$$

Vector $|0\rangle$ is not the zero vector; it is only the first vector in a collection of vectors. The zero vector is an exception, whose notation is not modified. Here we use the notation $\mathbf{0}$.

Suppose that vector $|v\rangle$ has the following entries in a basis

$$|v\rangle = \begin{bmatrix} a_1 \\ \vdots \\ a_n \end{bmatrix}.$$

The dual vector is denoted by $\langle v|$ and is defined by

$$\langle v| = \begin{bmatrix} a_1^* \cdots a_n^* \end{bmatrix}.$$

Vectors and their duals can be seen as column and row matrices, respectively. The matrix product of $\langle v|$ and $|v\rangle$, denoted by $\langle v|v\rangle$, is

$$\langle v|v\rangle = \sum_{i=1}^{n} a_i^* a_i,$$

which coincides with $(|v\rangle, |v\rangle)$. Then, the norm of a vector in the Dirac notation is

$$\| |v\rangle \| = \sqrt{\langle v|v\rangle}.$$

If $\{|v_1\rangle, \ldots, |v_n\rangle\}$ is an orthonormal basis, then

$$\langle v_i|v_j\rangle = \delta_{ij},$$

where δ_{ij} is the *Kronecker delta*. We use the terminology *ket* for the vector $|v\rangle$ and *bra* for the dual vector $\langle v|$. Keeping consistency, we use the terminology *bra–ket* for $\langle v|v\rangle$.

It is also very common to see the matrix product of $|v\rangle$ and $\langle v|$, denoted by $|v\rangle\langle v|$, known as the *outer product*, whose result is a $n \times n$ matrix

$$\begin{aligned} |v\rangle\langle v| &= \begin{bmatrix} a_1 \\ \vdots \\ a_n \end{bmatrix} \cdot \begin{bmatrix} a_1^* \cdots a_n^* \end{bmatrix} \\ &= \begin{bmatrix} a_1 a_1^* & \cdots & a_1 a_n^* \\ & \ddots & \\ a_n a_1^* & \cdots & a_n a_n^* \end{bmatrix}. \end{aligned}$$

The key to the Dirac notation is to always view *kets* as column matrices, *bras* as row matrices, and recognize that a sequence of *bras* and *kets* is a matrix product, hence associative, but noncommutative.

A.4 Computational Basis

The *computational basis* of $\mathbb{C}^n$ is $\{|0\rangle, \ldots, |n-1\rangle\}$, where

$$|0\rangle = \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \ldots, |n-1\rangle = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 1 \end{bmatrix}.$$

This basis is also known as *canonical basis*. A few times we use the numbering of the computational basis beginning with $|1\rangle$ and ending with $|n\rangle$. In this book, when we use a small-caption *Latin letter* within a *ket* or *bra*, we are referring to the computational basis. Then, the following expression is always valid

$$\langle i | j \rangle = \delta_{ij}.$$

The normalized sum of all computational basis vectors defines vector

$$|\mathrm{D}\rangle = \frac{1}{\sqrt{n}} \sum_{i=0}^{n-1} |i\rangle,$$

which we call *diagonal state*. When $n = 2$, the diagonal state is given by $|\mathrm{D}\rangle = |+\rangle$, where

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

Exercise A.3. Calculate explicitly the values of $|i\rangle\langle j|$ and

$$\sum_{i=0}^{n-1} |i\rangle\langle i|$$

in $\mathbb{C}^3$.

A.5 Qubit and the Bloch Sphere

The *qubit* is a *unit vector* in vector space $\mathbb{C}^2$. An arbitrary qubit $|\psi\rangle$ is represented by

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

where coefficients α and β are complex numbers and obey the constraint

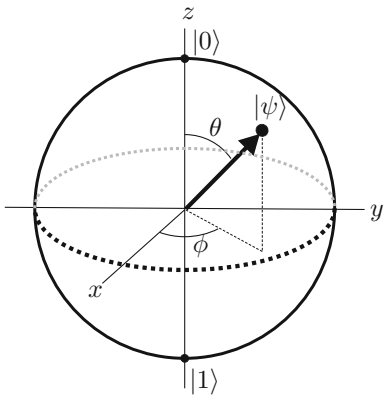


Fig. A.1 Bloch sphere. The state $|\psi\rangle$ of a qubit is represented by a point on the sphere

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

The set $\{|0\rangle,\ |1\rangle\}$ is the computational basis of $\mathbb{C}^2$, and $\alpha,\ \beta$ are called amplitudes of state $|\psi\rangle$. The term *state* (or *state vector*) is used as a synonym for *unit vector in a Hilbert space*.

In principle, we need four real numbers to describe a qubit, two for α and two for β. The constraint $|\alpha|^2 + |\beta|^2 = 1$ reduces to three numbers. In quantum mechanics, two vectors that differ from a *global phase factor* are considered equivalent. A global phase factor is a complex number of unit modulus multiplying the state. By eliminating this factor, a qubit can be described by two real numbers θ and ϕ as follows:

$$|\psi\rangle = \cos\frac{\theta}{2}\,|0\rangle + \mathrm{e}^{i\phi}\sin\frac{\theta}{2}\,|1\rangle,$$

where $0 \leq \theta \leq \pi$ and $0 \leq \phi < 2\pi$. In the notation above, state $|\psi\rangle$ can be represented by a point on the surface of a sphere of unit radius called *Bloch sphere*. Numbers θ and ϕ are spherical angles that locate the point that describes $|\psi\rangle$, as shown in Fig. A.1. The vector showed there is given by

$$\begin{bmatrix} \sin\theta\cos\phi \\ \sin\theta\sin\phi \\ \cos\theta \end{bmatrix}.$$

When we disregard global phase factors, there is a one-to-one correspondence between the quantum states of a qubit and the points on the Bloch sphere. State $|0\rangle$ is the *north pole* of the sphere because it is obtained by taking $\theta = 0$. State $|1\rangle$ is the *south pole*. States

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

are the intersection points of the x-axis and the sphere; states $(|0\rangle \pm \mathrm{i}|1\rangle)/\sqrt{2}$ are the intersection points of the y-axis with the sphere.

The representation of *classical bits* in this context is given by the poles of the Bloch sphere, and the representation of the *probabilistic classical bit*, that is, 0 with probability p and 1 with probability $1 - p$, is given by the point on z-axis with coordinate $2p - 1$. The interior of the Bloch sphere is used to describe states of a qubit in the presence of *decoherence*.

Exercise A.4. Using the Dirac notation, show that opposite points on the Bloch sphere correspond to orthogonal states.

Exercise A.5. Suppose you know that the state of a qubit is either $|+\rangle$ with probability p or $|-\rangle$ with probability $1 - p$. If this is the best you know about the state of the qubit, where on the Bloch sphere would you represent this qubit?

A.6 Linear Operators

Let V and W be vector spaces, $\{|v_1\rangle, \ldots, |v_n\rangle\}$ a basis for V, and $\mathcal{A}$ a function $\mathcal{A} : V \mapsto W$ that satisfies

$$\mathcal{A}\left(\sum_i a_i |v_i\rangle\right) = \sum_i a_i \mathcal{A}(|v_i\rangle),$$

for any complex numbers a_i. $\mathcal{A}$ is called a *linear operator* from V to W. The term *linear operator on* V means that both the domain and codomain of $\mathcal{A}$ are V. The composition of linear operators $\mathcal{A} : V_1 \mapsto V_2$ and $\mathcal{B} : V_2 \mapsto V_3$ is also a linear operator $\mathcal{C} : V_1 \mapsto V_3$ obtained through the composition of their functions: $\mathcal{C}(|v\rangle) = \mathcal{B}(\mathcal{A}(|v\rangle))$. The sum of two linear operators, both from V to W, is defined by formula $(\mathcal{A} + \mathcal{B})(|v\rangle) = \mathcal{A}(|v\rangle) + \mathcal{B}(|v\rangle)$.

The *identity operator* $\mathcal{I}$ on V is a linear operator such that $\mathcal{I}(|v\rangle) = |v\rangle$ for all $|v\rangle \in V$. The *null operator* $\mathcal{O}$ on V is a linear operator such that $\mathcal{O}(|v\rangle) = \mathbf{0}$ for all $|v\rangle \in V$.

The *rank* of a linear operator $\mathcal{A}$ on V is the dimension of the image of $\mathcal{A}$. The *kernel* or *nullspace* or *support* of a linear operator $\mathcal{A}$ on V is the set of all vectors $|v\rangle$ such that $\mathcal{A}(|v\rangle) = \mathbf{0}$. The dimension of the kernel is called the *nullity* of the operator. The *rank–nullity theorem* states that $\mathrm{rank}(\mathcal{A}) + \mathrm{nullity}(\mathcal{A}) = \dim(V)$.

Fact

If we specify the action of a linear operator $\mathcal{A}$ on a basis of vector space V, the action of $\mathcal{A}$ on any vector in V can be determined by using the linearity property.

A.7 Matrix Representation

Linear operators are represented by matrices. Let $\mathcal{A} : V \mapsto W$ be a linear operator. Let $\{|v_1\rangle, \ldots, |v_n\rangle\}$ and $\{|w_1\rangle, \ldots, |w_m\rangle\}$ be orthonormal bases for V and W, respectively. The *matrix representation* of $\mathcal{A}$ is obtained by applying $\mathcal{A}$ to every vector in the basis of V and expressing the result as a linear combination of basis vectors of W, as follows:

$$\mathcal{A}\left(|v_j\rangle\right) = \sum_{i=1}^{m} a_{ij}|w_i\rangle,$$

where index j run from 1 to n. Therefore, a_{ij} are entries of a $m \times n$ matrix, which we call A. In this case, expression $\mathcal{A}\left(|v_j\rangle\right)$, which means function $\mathcal{A}$ applied to argument $|v_j\rangle$, is equivalent to the matrix product $A|v_j\rangle$. Using the outer product notation, we have

$$A = \sum_{i=1}^{m}\sum_{j=1}^{n} a_{ij}|w_i\rangle\langle v_j|.$$

Using the above equation and the fact that the basis of V is orthonormal, we can verify that the matrix product of A and $|v_j\rangle$ is equal to $\mathcal{A}\left(|v_j\rangle\right)$. The key to this calculation is to use the associativity of matrix multiplication:

$$\begin{aligned}\left(|w_i\rangle\langle v_j|\right)|v_k\rangle &= |w_i\rangle\left(\langle v_j|v_k\rangle\right)\\ &= \delta_{jk}|w_i\rangle.\end{aligned}$$

In particular, the matrix representation of the identity operator $\mathcal{I}$ in any orthonormal basis is the identity matrix I and the matrix representation of the null operator $\mathcal{O}$ in any orthonormal basis is the *zero matrix*.

If the linear operator $\mathcal{C}$ is the composition of the linear operators $\mathcal{B}$ and $\mathcal{A}$, the matrix representation of $\mathcal{C}$ is obtained by multiplying the matrix representation of $\mathcal{B}$ with that of $\mathcal{A}$, that is, $C = BA$.

When we fix orthonormal bases for the vector spaces, there is a one-to-one correspondence between linear operators and matrices. In $\mathbb{C}^n$, we use the computational basis as a reference basis, so that the terms *linear operator* and *matrix* are taken as synonyms. We also use the term *operator* as a synonym for *linear operator*.

Exercise A.6. Suppose B is an operator whose action on the computational basis of the n-dimensional vector space V is

$$B|j\rangle = |\psi_j\rangle,$$

where $|\psi_j\rangle$ are vectors in V for all j.

1. Obtain the expression of B using the outer product.
2. Show that $|\psi_j\rangle$ is the jth column in the matrix representation of B.

3. Suppose that B is the Hadamard operator

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

Redo the previous items using operator H.

A.8 Diagonal Representation

Let A be an operator on V. If there exists an orthonormal basis $\{|v_1\rangle, \ldots, |v_n\rangle\}$ of V such that

$$A = \sum_{i=1}^{n} \lambda_i |v_i\rangle\langle v_i|,$$

we say that A admits a *diagonal representation* or, equivalently, A is *diagonalizable*. The complex numbers λ_i are the *eigenvalues* of A and $|v_i\rangle$ are the corresponding *eigenvectors*. A vector $|\psi\rangle$ is an *eigenvector* of A if there is a scalar λ, called *eigenvalue*, so that

$$A|\psi\rangle = \lambda|\psi\rangle.$$

Any multiple of an eigenvector is also an eigenvector. If two eigenvectors are associated with the same eigenvalue, then any linear combination of these eigenvectors is an eigenvector. The number of linearly independent eigenvectors associated with the same eigenvalue is the *multiplicity* of that eigenvalue. We use the short notation "λ-eigenvectors" for eigenvectors associated with eigenvalues λ.

If there are eigenvalues with multiplicity greater than one, the diagonal representation is factored out as follows:

$$A = \sum_{\lambda} \lambda P_\lambda,$$

where index λ runs only on the distinct eigenvalues and P_λ is the projector on the eigenspace of A associated with eigenvalue λ. If λ has multiplicity 1, $P_\lambda = |v\rangle\langle v|$, where $|v\rangle$ is the unit eigenvector associated with λ. If λ has multiplicity 2 and $|v_1\rangle, |v_2\rangle$ are linearly independent unit eigenvectors associated with λ, $P_\lambda = |v_1\rangle\langle v_1| + |v_2\rangle\langle v_2|$ and so on. The projectors P_λ satisfy

$$\sum_{\lambda} P_\lambda = I.$$

An alternative way to define a diagonalizable operator is by requiring that A is *similar* to a diagonal matrix. Matrices A and A' are similar if $A' = M^{-1}AM$ for some invertible matrix M. We have interest only in the case when M is a unitary matrix.

The term *diagonalizable* used here is narrower than the one used in the literature because we are demanding that M be a unitary matrix.

The *characteristic polynomial* of a matrix A, denoted by $p_A(\lambda)$, is the monic polynomial

$$p_A(\lambda) = \det(\lambda I - A).$$

The roots of $p_A(\lambda)$ are the eigenvalues of A. Usually, the best way to calculate the eigenvalues of a matrix is via the characteristic polynomial. For a two-dimensional matrix U, the characteristic polynomial is given by

$$p_U(\lambda) = \lambda^2 - \mathrm{tr}(U)\,\lambda + \det(U).$$

If U is a real unitary matrix, the eigenvalues have the form $\mathrm{e}^{\pm \mathrm{i}\omega}$ and the characteristic polynomial is given by

$$p_U(\lambda) = \lambda^2 - 2\lambda\cos\omega + 1.$$

Exercise A.7. Suppose that A is a diagonalizable operator with eigenvalues ± 1. Show that

$$P_{\pm 1} = \frac{I \pm A}{2}.$$

A.9 Completeness Relation

The *completeness relation* is so useful that it deserves to be highlighted. Let $\{|v_1\rangle, \ldots, |v_n\rangle\}$ be an orthonormal basis of V. Then,

$$I = \sum_{i=1}^{n} |v_i\rangle\langle v_i|.$$

The completeness relation is the diagonal representation of the identity matrix.

Exercise A.8. If $\{|v_1\rangle, \ldots, |v_n\rangle\}$ is an orthonormal basis, it is straightforward to show that

$$A = \sum_{i=1}^{m}\sum_{j=1}^{n} a_{ij}|w_i\rangle\langle v_j|$$

implies

$$A|v_j\rangle = \sum_{i=1}^{m} a_{ij}|w_i\rangle.$$

Prove the reverse, that is, given the above expressions for $A|v_j\rangle$, use the completeness relation to obtain A. [Hint: Multiply the last equation by $\langle v_j|$ and sum over j.]

A.10 Cauchy–Schwarz Inequality

Let V be a Hilbert space and $|v\rangle, |w\rangle \in V$. Then,

$$\left|\langle v|w\rangle\right| \leq \sqrt{\langle v|v\rangle\langle w|w\rangle}.$$

A more explicit way of presenting the Cauchy–Schwarz inequality is

$$\left|\sum_i v_i w_i\right|^2 \leq \left(\sum_i |v_i|^2\right)\left(\sum_i |w_i|^2\right),$$

which is obtained when we take $|v\rangle = \sum_i v_i^* |i\rangle$ and $|w\rangle = \sum_i w_i |i\rangle$.

A.11 Special Operators

Let A be a linear operator on Hilbert space V. Then, there exists a unique linear operator $A^\dagger$ on V, called *adjoint operator*, that satisfies

$$(|v\rangle, A|w\rangle) = \left(A^\dagger|v\rangle, |w\rangle\right),$$

for all $|v\rangle, |w\rangle \in V$.

The matrix representation of $A^\dagger$ is the transpose–conjugate matrix $(A^*)^T$. The main properties of the *dagger* or *transpose–conjugate* operation are

1. $(A\,B)^\dagger = B^\dagger A^\dagger$
2. $|v\rangle^\dagger = \langle v|$
3. $\left(A|v\rangle\right)^\dagger = \langle v|A^\dagger$
4. $\left(|w\rangle\langle v|\right)^\dagger = |v\rangle\langle w|$
5. $\left(A^\dagger\right)^\dagger = A$
6. $\left(\sum_i a_i A_i\right)^\dagger = \sum_i a_i^* A_i^\dagger$

The last property shows that the dagger operation is *conjugate-linear* when applied on a linear combination of operators.

Normal Operator

An operator A on V is *normal* if $A^\dagger A = A A^\dagger$.

Spectral Theorem

An operator A on V is diagonalizable if and only if A is normal.

Unitary Operator

An operator U on V is *unitary* if $U^\dagger U = U U^\dagger = I$.

Facts about Unitary Operators

Unitary operators are normal. They are diagonalizable with respect to an orthonormal basis. Eigenvectors of a unitary operator associated with different eigenvalues are orthogonal. The eigenvalues have unit modulus, that is, they have the form $e^{i\alpha}$, where α is a real number . Unitary operators preserve the inner product, that is, the inner product of $U|v_1\rangle$ and $U|v_2\rangle$ is equal to the inner product of $|v_1\rangle$ and $|v_2\rangle$. The action of a unitary operator on a vector preserves its norm.

Hermitian Operator

An operator A on V is *Hermitian* or *self-adjoint* if $A^\dagger = A$.

Facts about Hermitian Operators

Hermitian operators are normal. They are diagonalizable with respect to an orthonormal basis. Eigenvectors of a Hermitian operator associated with different eigenvalues are orthogonal. The eigenvalues of a Hermitian operator are real numbers. A real symmetric matrix is Hermitian.

Orthogonal Projector

An operator P on V is an *orthogonal projector* if $P^2 = P$ and $P^\dagger = P$.

Facts about Orthogonal Projectors

The eigenvalues are equal to 0 or 1. If P is an orthogonal projector, then the *orthogonal complement* $I - P$ is also an orthogonal projector. Applying a projector to a vector either decreases its norm or maintains invariant. In this book, we use the term *projector* as a synonym for *orthogonal projector*. We use the term *nonorthogonal projector* explicitly to distinguish this case. An example of a nonorthogonal projector on a qubit is $P = |1\rangle(\langle 0| + \langle 1|)$. Note that P is not normal in this example.

Positive Operator

An operator A on V is said *positive* if $\langle v|A|v\rangle \geq 0$ for any $|v\rangle \in V$. If the inequality is strict for any nonzero vector in V, then the operator is said *positive definite*.

Facts about Positive Operators

Positive operators are Hermitian. The eigenvalues are nonnegative real numbers.

Exercise A.9. Consider matrix

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

1. Show that M is not normal.
2. Show that the eigenvectors of M generate a one-dimensional space.

Exercise A.10. Consider matrix

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

1. Show that the eigenvalues of M are ± 1.
2. Show that M is not unitary nor Hermitian.
3. Show that the eigenvectors associated with distinct eigenvalues of M are not orthogonal.
4. Show that M has a diagonal representation.

Exercise A.11. 1. Show that the product of two unitary operators is a unitary operator.
2. The sum of two unitary operators is necessarily a unitary operator? If not, give a counterexample.

Exercise A.12. 1. Show that the sum of two Hermitian operators is a Hermitian operator.
2. The product of two Hermitian operators is necessarily a Hermitian operator? If not, give a counterexample.

Exercise A.13. Show that $A^\dagger A$ is a positive operator for any operator A.

A.12 Pauli Matrices

The *Pauli matrices* are

$$\sigma_0 = I = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix},$$

$$\sigma_1 = \sigma_x = X = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix},$$

$$\sigma_2 = \sigma_y = Y = \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix},$$

$$\sigma_3 = \sigma_z = Z = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}.$$

These matrices are unitary and Hermitian, and hence their eigenvalues are equal to ± 1. Putting in another way: $\sigma_j^2 = I$ and $\sigma_j^\dagger = \sigma_j$ to $j = 0, \ldots, 3$.

The following facts are extensively used:

$$X|0\rangle = |1\rangle, \; X|1\rangle = |0\rangle,$$

$$Z|0\rangle = |0\rangle, \;\; Z|1\rangle = -|1\rangle.$$

Pauli matrices form a basis for the vector space of 2×2 matrices. Therefore, an arbitrary operator that acts on a qubit can be written as a linear combination of Pauli matrices.

Exercise A.14. Consider the representation of the state $|\psi\rangle$ of a qubit on the Bloch sphere. What is the representation of states $X|\psi\rangle$, $Y|\psi\rangle$, and $Z|\psi\rangle$ relative to $|\psi\rangle$? What is the geometric interpretation of the action of the Pauli matrices on the Bloch sphere?

A.13 Operator Functions

If we have an operator A on V, we ask whether it is possible to calculate $\sqrt{A}$, that is, to find an operator whose square is A? It is more interesting to ask ourselves whether it makes sense to use an operator as an argument of an arbitrary function $f : \mathbb{C} \mapsto \mathbb{C}$, such as the exponential or logarithmic function. If f is analytic, we use the Taylor expansion of $f(x)$ and replace x by A. This will not work for the square root function. There is an alternate route for lifting f if operator A is normal. Using the diagonal representation, A can be written in the form

$$A = \sum_i a_i |v_i\rangle\langle v_i|,$$

where a_i are the eigenvalues and the set $\{|v_i\rangle\}$ is an orthonormal basis of eigenvectors of A. We extend the application of a function $f : \mathbb{C} \mapsto \mathbb{C}$ to the set of normal operators as follows. If A is a normal operator, then

$$f(A) = \sum_i f(a_i)|v_i\rangle\langle v_i|.$$

The result is an operator defined on the same vector space V.

If the goal is to calculate $\sqrt{A}$, first A must be diagonalized, that is, we must determine a unitary matrix U such that $A = U D U^\dagger$, where D is a diagonal matrix. Then, we use the fact that $\sqrt{A} = U\sqrt{D}\,U^\dagger$, where $\sqrt{D}$ is calculated by taking the square root of each diagonal element.

If U is the evolution operator of an isolated quantum system whose state is $|\psi(0)\rangle$ initially, the state at time t is given by

$$|\psi(t)\rangle \; = \; U^t|\psi(0)\rangle.$$

Usually, the most efficient way to calculate state $|\psi(t)\rangle$ is to obtain the diagonal representation of the unitary operator U, described as

$$U = \sum_i \lambda_i |v_i\rangle\langle v_i|,$$

and to calculate the tth power of U, which is

$$U^t = \sum_i \lambda_i^t |v_i\rangle\langle v_i|.$$

The system state at time t will be

$$|\psi(t)\rangle = \sum_i \lambda_i^t \langle v_i|\psi(0)\rangle |v_i\rangle.$$

The *trace* of a matrix is another type of operator function. In this case, the result of applying the trace function is a complex number defined as

$$\mathrm{tr}(A) = \sum_i a_{ii},$$

where a_{ii} is the ith diagonal element of A. In the Dirac notation,

$$\mathrm{tr}(A) = \sum_i \langle v_i|A|v_i\rangle,$$

where $\{|v_1\rangle, \ldots, |v_n\rangle\}$ is an orthonormal basis of V. The trace function satisfies the following properties:

1. (Linearity) $\mathrm{tr}(aA + bB) = a\,\mathrm{tr}(A) + b\,\mathrm{tr}(B)$,
2. (Commutativity) $\mathrm{tr}(AB) = \mathrm{tr}(BA)$,
3. (Cyclic property) $\mathrm{tr}(A\,B\,C) = \mathrm{tr}(C\,A\,B)$.

The third property follows from the second one. Properties 2 and 3 are valid when A, B, and C are not square matrices (AB, ABC, and CAB must be square matrices).

The trace function is invariant under the *similarity transformation*, that is, $\mathrm{tr}(M^{-1}AM) = \mathrm{tr}(A)$, where M is an invertible matrix. This implies that the trace does not depend on the basis choice for the matrix representation of A.

A useful formula involving the trace of operators is

$$\mathrm{tr}(A|\psi\rangle\langle\psi|) = \langle\psi|A|\psi\rangle,$$

for any $|\psi\rangle \in V$ and any A on V. This formula is easily proved using the cyclic property of the trace function.

Exercise A.15. Using the method of applying functions to matrices described in this section, find all matrices M such that

$$M^2 = \begin{bmatrix} 5 & 4 \\ 4 & 5 \end{bmatrix}.$$

Exercise A.16. If f is analytic and A is normal, show that $f(A)$ using the Taylor expansion is equal to $f(A)$ using the spectral decomposition.

A.14 Norm of a Linear Operator

Given a vector space V over the complex numbers, a *norm* on V is a function $\| \| : V \to \mathbb{R}$ with the following properties:

- $\|a|\psi\rangle\| = |a| \, \||\psi\rangle\|$,
- $\||\psi\rangle + |\psi'\rangle\| \leq \||\psi\rangle\| + \||\psi'\rangle\|$,
- $\||\psi\rangle\| \geq 0$,
- $\||\psi\rangle\| = 0$ if and only if $|\psi\rangle = \mathbf{0}$,

for all $a \in \mathbb{C}$ and all $|\psi\rangle, |\psi'\rangle \in V$,

The set of all linear operators on a Hilbert space $\mathcal{H}$ is a vector space over the complex numbers because it obeys the properties demanded by the definition described in Sect. A.1. It is possible to define more than one norm on a vector space, and let us start with the following norm.

Let A be a linear operator on a Hilbert space $\mathcal{H}$. The *norm* of A is defined as

$$\|A\| = \max_{\langle\psi|\psi\rangle=1} \left|\langle\psi|A|\psi\rangle\right|,$$

where the maximum is over all normalized states $|\psi\rangle \in \mathcal{H}$.

The next norm is induced from an *inner product*. The *Hilbert–Schmidt inner product* (also known as *Frobenius inner product*) of two linear operators A and B is

$$(A, B) = \mathrm{tr}\left(A^\dagger B\right).$$

Now, we can define another norm (*trace norm*) of a linear operator A on a Hilbert space $\mathcal{H}$ as

$$\|A\|_{\mathrm{tr}} = \sqrt{\mathrm{tr}\left(A^\dagger A\right)}.$$

In a *normed vector space*, the *distance* between vectors $|\psi\rangle$ and $|\psi'\rangle$ is given by $\||\psi\rangle - |\psi'\rangle\|$. Then, it makes sense to speak about *distance between linear operators* A and B, which is the nonnegative number $\|A - B\|$.

Exercise A.17. Show that $\|U\| = 1$ if U is a unitary operator.

Exercise A.18. Show that the inner product $(A, B) = \mathrm{tr}(A^\dagger B)$ satisfies the properties described in Sect. A.2.

Exercise A.19. Show that $\|U\|_{\text{tr}} = \sqrt{n}$ if U is a unitary operator on $\mathbb{C}^n$.

Exercise A.20. Show that both norms defined in this section satisfy the properties described at the beginning of this section.

A.15 Tensor Product

Let V and W be finite Hilbert spaces with basis $\{|v_1\rangle, \ldots, |v_m\rangle\}$ and $\{|w_1\rangle, \ldots, |w_n\rangle\}$, respectively. The *tensor product* of V and W, denoted by $V \otimes W$, is a (mn)-dimensional Hilbert space with basis $\{|v_1\rangle \otimes |w_1\rangle, |v_1\rangle \otimes |w_2\rangle, \ldots, |v_m\rangle \otimes |w_n\rangle\}$. The tensor product of a vector in V and a vector in W, such as $|v\rangle \otimes |w\rangle$, also denoted by $|v\rangle|w\rangle$ or $|v, w\rangle$ or $|v\, w\rangle$, is calculated explicitly via the Kronecker product, defined ahead. An arbitrary vector in $V \otimes W$ is a linear combination of vectors $|v_i\rangle \otimes \big|w_j\big\rangle$, that is, if $|\psi\rangle \in V \otimes W$, then

$$|\psi\rangle = \sum_{i=1}^{m} \sum_{j=1}^{n} a_{ij} |v_i\rangle \otimes \big|w_j\big\rangle.$$

The tensor product is *bilinear*, that is, linear with respect to each argument:

1. $|v\rangle \otimes \big(a\,|w_1\rangle + b\,|w_2\rangle\big) \;=\; a\,|v\rangle \otimes |w_1\rangle + b\,|v\rangle \otimes |w_2\rangle,$
2. $\big(a\,|v_1\rangle + b\,|v_2\rangle\big) \otimes |w\rangle \;=\; a\,|v_1\rangle \otimes |w\rangle + b\,|v_2\rangle \otimes |w\rangle.$

A scalar can always be factored out to the beginning of the expression:

$$a\big(|v\rangle \otimes |w\rangle\big) \;=\; \big(a|v\rangle\big) \otimes |w\rangle \;=\; |v\rangle \otimes \big(a|w\rangle\big).$$

The tensor product of a linear operator A on V and B on W, denoted by $A \otimes B$, is a linear operator on $V \otimes W$ defined by

$$\big(A \otimes B\big)\big(|v\rangle \otimes |w\rangle\big) \;=\; \big(A|v\rangle\big) \otimes \big(B|w\rangle\big).$$

In general, an arbitrary linear operator on $V \otimes W$ cannot be factored out as the tensor product of the form $A \otimes B$, but it can be written as a linear combination of operators of the form $A_i \otimes B_j$. The above definition is easily extended to operators $A : V \mapsto V'$ and $B : W \mapsto W'$. In this case, the tensor product of these operators is $(A \otimes B) : (V \otimes W) \mapsto (V' \otimes W')$.

In quantum mechanics, it is very common to use operators in the form of external products, for example, $A = |v\rangle\langle v|$ and $B = |w\rangle\langle w|$. The tensor product of A and B is represented by the following equivalent ways:

$$\begin{aligned} A \otimes B &= \big(|v\rangle\langle v|\big) \otimes \big(|w\rangle\langle w|\big) \\ &= |v\rangle\langle v| \otimes |w\rangle\langle w| \\ &= |v, w\rangle\langle v, w|. \end{aligned}$$

If A_1, A_2 are operators on V and B_1, B_2 are operators on W, then

$$(A_1 \otimes B_1) \cdot (A_2 \otimes B_2) = (A_1 \cdot A_2) \otimes (B_1 \cdot B_2).$$

The inner product of $|v_1\rangle \otimes |w_1\rangle$ and $|v_2\rangle \otimes |w_2\rangle$ is defined as

$$\big(|v_1\rangle \otimes |w_1\rangle\,,\ |v_2\rangle \otimes |w_2\rangle\big) \ = \ \big\langle v_1 \big| v_2\big\rangle \big\langle w_1 \big| w_2\big\rangle.$$

The inner product of vectors written as a linear combination of basis vectors is calculated by applying the linear property to the second argument and the *conjugate-linear* property on the first argument of the inner product. For example,

$$\left(\left(\sum_{i=1}^{n} a_i |v_i\rangle\right) \otimes |w_1\rangle\,,\ |v\rangle \otimes |w_2\rangle\right) \ = \ \left(\sum_{i=1}^{n} a_i^* \big\langle v_i \big| v\big\rangle\right) \big\langle w_1 \big| w_2\big\rangle.$$

The inner product definition implies that

$$\big\|\, |v\rangle \otimes |w\rangle \,\big\| \ = \ \big\|\, |v\rangle \,\big\| \cdot \big\|\, |w\rangle \,\big\|.$$

In particular, the tensor product of unit norm vectors is a unit norm vector.

When we use matrix representations of operators, the tensor product is calculated explicitly via the *Kronecker product*. Let A be a $m \times n$ matrix and B a $p \times q$ matrix. Then,

$$A \otimes B = \begin{bmatrix} a_{11}B & \cdots & a_{1n}B \\ & \ddots & \\ a_{m1}B & \cdots & a_{mn}B \end{bmatrix}.$$

The dimension of the resulting matrix is $mp \times nq$. The Kronecker product is used for matrices of any dimension, particularly for two vectors,

$$\begin{bmatrix} a_1 \\ a_2 \end{bmatrix} \otimes \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} = \begin{bmatrix} a_1 \begin{bmatrix} b_1 \\ \\ b_2 \end{bmatrix} \\ \\ a_2 \begin{bmatrix} b_1 \\ \\ b_2 \end{bmatrix} \end{bmatrix} = \begin{bmatrix} a_1 b_1 \\ \\ a_1 b_2 \\ \\ a_2 b_1 \\ \\ a_2 b_2 \end{bmatrix}.$$

The tensor product is an associative and distributive operation, but noncommutative, that is, $|v\rangle \otimes |w\rangle \neq |w\rangle \otimes |v\rangle$ if $v \neq w$. Most operations on a tensor product are performed term by term, such as

$$(A \otimes B)^\dagger = A^\dagger \otimes B^\dagger.$$

If both operators A and B are special operators of the same type, as the ones defined in Sect. A.11, then the tensor product $A \otimes B$ is also a special operator of the same type. For example, the tensor product of Hermitian operators is a Hermitian operator.

The trace of a Kronecker product of matrices is

$$\mathrm{tr}(A \otimes B) = \mathrm{tr}A \ \mathrm{tr}B,$$

while the determinant is

$$\det(A \otimes B) = (\det A)^m \ (\det B)^n,$$

where n is the dimension of A and m of B.

If the *diagonal state* of the vector space V is $|\mathrm{D}\rangle_V$ and of space W is $|\mathrm{D}\rangle_W$, then the diagonal state of space $V \otimes W$ is $|\mathrm{D}\rangle_V \otimes |\mathrm{D}\rangle_W$. Therefore, the diagonal state of space $V^{\otimes n}$ is $|\mathrm{D}\rangle^{\otimes n}$, where $V^{\otimes n}$ means $V \otimes \cdots \otimes V$ with n terms.

Exercise A.21. Let H be the Hadamard operator

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

Show that

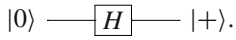
$$\langle i|H^{\otimes n}|j\rangle = \frac{(-1)^{i \cdot j}}{\sqrt{2^n}},$$

where n represents the number of qubits and $i \cdot j$ is the binary inner product, that is, $i \cdot j = i_1 j_1 + \cdots + i_n j_n \mod 2$, where $(i_1, \ldots, i_n)$ and $(j_1, \ldots, j_n)$ are the binary decompositions of i and j, respectively.

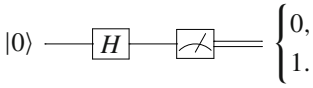
A.16 Quantum Gates, Circuits, and Registers

A *quantum circuit* is a pictorial way to describe a *quantum algorithm*. The input lies on the left-hand side of the circuit and the *quantum information* flows unchanged through the wires, from the left to right, until finding a *quantum gate*, which is a square box with the name of a unitary operator. The quantum gate represents the action of the unitary operator, which transforms the quantum information and releases it to the wire on the right-hand side. For example, the algebraic expression $|+\rangle = H|0\rangle$

is represented by the circuit[1]



If a measurement is performed, the representation is

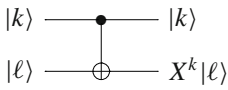


A *meter* represents a measurement in the computational basis, and a double wire conveys the *classical information* that comes out of the meter. In the example, the state of the qubit right before the measurement is $(|0\rangle + |1\rangle)/\sqrt{2}$. If the qubit state is projected on $|0\rangle$ after the measurement, the output is 0, otherwise 1.

The *controlled NOT gate* (CNOT or $C(X)$) is a 2-qubit gate defined by

$$\text{CNOT}\,|k\rangle|\ell\rangle = |k\rangle X^k|\ell\rangle,$$

and represented by the circuit

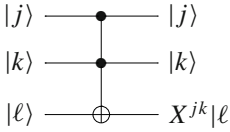


The qubit marked with the black full point is called *control qubit*, and the qubit marked with the $\oplus$ sign is called the *target qubit*.

The *Toffoli gate* $C^2(X)$ is a 3-qubit controlled gate defined by

$$C^2(X)\,|j\rangle|k\rangle|\ell\rangle = |j\rangle|k\rangle X^{jk}|\ell\rangle,$$

and represented by the circuit



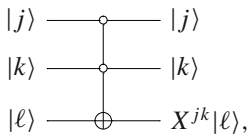
The Toffoli gate has two control qubits and one target qubit.

These gates can be generalized. The *generalized Toffoli gate* $C^n(X)$ is a $(n+1)$-qubit controlled gate defined by

$$C^n(X)\,|j_1\rangle...|j_n\rangle|j_{n+1}\rangle = |j_1\rangle...|j_n\rangle X^{j_1\cdots j_n}|j_{n+1}\rangle.$$

When defined using the computational basis, the state of the target qubit inverts if and only if all control qubits are set to one. There is another case in which the state of the target qubit inverts if and only if all control qubits are set to zero. In this case, the control qubits are depicted by *empty controls* (empty white points) instead of *full controls* (full black points), such as

[1]The circuits were generated with package Q-circuit.



whose algebraic representation is

$$|j_1\rangle|j_2\rangle|j_3\rangle \longmapsto |j_1\rangle|j_2\rangle X^{(1-j_1)(1-j_2)}|j_3\rangle.$$

It is possible to mix full and empty controls. Those kinds of controlled gates are called *generalized Toffoli gates*.

A *register* is a set of qubits treated as a composite system. In many quantum algorithms, the qubits are divided into two registers: one for the main calculation from where the result comes out and one for the draft (calculations that will be discarded).

Suppose we have a register with two qubits. The computational basis is

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

An arbitrary state of this register is

$$|\psi\rangle = \sum_{i=0}^{1}\sum_{j=0}^{1} a_{ij}|i,j\rangle$$

where coefficients a_{ij} are complex numbers that satisfy the constraint

$$|a_{00}|^2 + |a_{01}|^2 + |a_{10}|^2 + |a_{11}|^2 = 1.$$

To help to generalize to n qubits, it is usual to compress the notation by converting the base-2 notation to the base-10 notation. The computational basis for a two-qubit register in the base-10 notation is $\{|0\rangle, |1\rangle, |2\rangle, |3\rangle\}$. In the base-2 notation, we can determine the number of qubits by counting the number of digits inside the *ket*; for example, $|011\rangle$ refers to three qubits. In the base-10 notation, we cannot determine what is the number of qubits of the register. The number of qubits is implicit. At any point, we can go back, write the numbers in the base-2 notation, and the number of qubits will be clear. In the compact notation, an arbitrary state of a n-qubit register is

$$|\psi\rangle = \sum_{i=0}^{2^n-1} a_i|i\rangle$$

where coefficients a_i are complex numbers that satisfy the constraint

$$\sum_{i=0}^{2^n-1} |a_i|^2 = 1.$$

The *diagonal state* of a n-qubit register is the tensor product of the diagonal state of each qubit, that is, $|\mathrm{D}\rangle = |+\rangle^{\otimes n}$.

A set of *universal quantum gates* is a finite set of gates that generates any unitary operator through tensor and matrix products of gates in the set. Since the number of possible quantum gates is uncountable even in the 1-qubit case, we require that any quantum gate can be approximated by a sequence of universal quantum gates. One simple set of *universal gates* is CNOT, H, X, T (or $\pi/8$ gate), and $T^\dagger$, where

$$T = \begin{bmatrix} 1 & 0 \\ 0 & \mathrm{e}^{\frac{\mathrm{i}\pi}{4}} \end{bmatrix}.$$

To calculate the *time complexity* of a quantum algorithm, we have to implement a circuit of the algorithm in terms of universal gates in the best way possible. The time complexity is determined by the depth of the circuit.

For instance, Fig. A.2 shows the decomposition of the Toffoli gate into universal gates. The right-hand circuit has only universal gates (15 gates) and depth 12.

A Toffoli gate with empty controls can be decomposed in terms of a standard Toffoli gate and X gates as depicted in the right-hand circuit of Fig. A.3.

Figure A.4 shows the decomposition of a 6-qubit generalized Toffoli gate with five full controls into Toffoli gates. If the generalized Toffoli gate has n controls, we use $(n-2)$ *ancilla*[2] qubits initially in state $|0\rangle$. The ancilla qubits are interlaced with the controls starting from the second control.

Exercise A.22. Show that the diagonal state of a n-qubit register is $|\mathrm{D}\rangle = |+\rangle^{\otimes n}$ or equivalently $|\mathrm{D}\rangle = H^{\otimes n}|0,\ldots,0\rangle$.

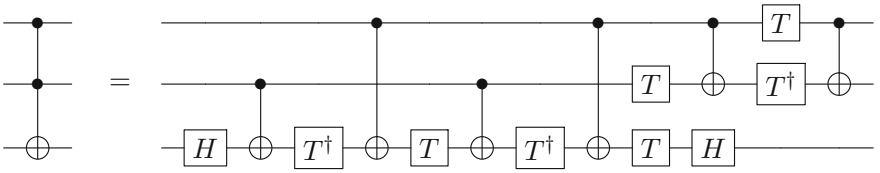
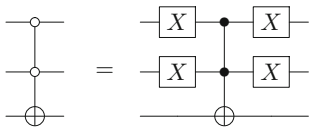


Fig. A.2 Decomposition of a Toffoli gates into universal gates

Fig. A.3 Converting empty controls into full controls



[2] Ancilla means auxiliary.

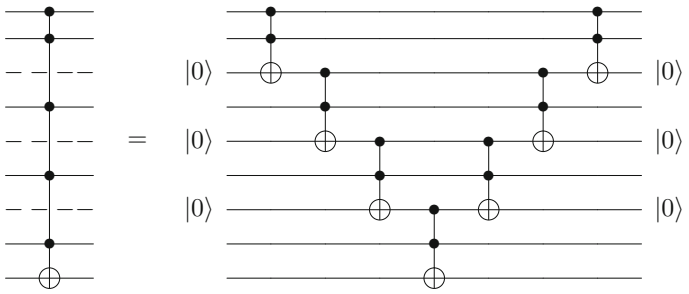


Fig. A.4 Decomposition of a generalized 6-qubit Toffoli gate into Toffoli gates. This decomposition can be easily extended for generalized Toffoli gates with any number of control qubits

Exercise A.23. Let f be a function with domain $\{0,1\}^n$ and codomain $\{0,1\}^m$. Consider a 2-register quantum computer with n and m qubits, respectively. Function f can be implemented by using operator U_f defined in the following way:

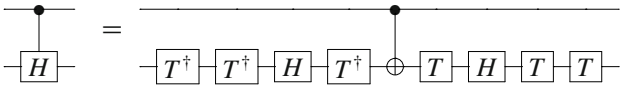
$$U_f|x\rangle|y\rangle = |x\rangle|y \oplus f(x)\rangle,$$

where x has n bits, y has m bits, and $\oplus$ is the binary sum (bitwise *xor*).

1. Show that U_f is a unitary operator for any f.
2. If $n = m$ and f is injective, show that f can be implemented on a 1-register quantum computer with n qubits.

Exercise A.24. Show that the circuits of Fig. A.4 are equivalent. Find the number of universal gates in the decomposition of a generalized Toffoli gate with n_1 empty controls and n_2 full controls. Find the depth of the circuit.

Exercise A.25. Show that the controlled Hadamard $C(H)$ can be decomposed into universal gates as depicted in the following circuit.



Further Reading

There are many good books on linear algebra. For an initial contact, Refs. [24, 28, 200, 305] are good options; for a more advanced approach, Refs. [148, 150, 199] are good options; for those who have mastered the basics and are only interested in the application of linear algebra to quantum computation, Ref. [248] is recommended, especially for the decomposition of unitary gates into *universal gates*. Linear algebra for quantum algorithms is addressed in [209]. The *Dirac notation* is clearly and comprehensively presented in [287].

Appendix B
Graph Theory for Quantum Walks

Graph theory is a large area of mathematics with a wide range of applications, especially in computer science. It is impossible to overstate the importance of graph theory for quantum walks. In fact, graph theory for quantum walks is as important as linear algebra for quantum computation.

In the quantum walk setting, the graph represents positions and directions for the walker's shift. It is not mandatory to use the graph vertices as the walker's position. Any interpretation is accepted if it employs the graph structure so that the physical meaning reflects the graph components. For example, it makes no sense to have a quantum walk model in which the walker can jump over some vertices, for instance, a model on the line in which the walker can jump from vertex 1 to vertex 3, skipping vertex 2. If it is allowed to jump from vertex 1 to vertex 3, it means that there is an edge or arc linking vertex 1 to vertex 3 and the underlying graph is not the line.

A solid basis on graph theory is required to understand the area of quantum walk. This appendix focuses on the main definitions of graph theory used in this work with some brief examples and should not be used as the first contact with graph theory. At the end of this appendix, introductory and advanced references for starters and for further reading are given.

B.1 Basic Definitions

A *simple graph* $\Gamma(V, E)$ is defined by a set $V(\Gamma)$ of vertices or nodes and a set $E(\Gamma)$ of edges so that each edge links two vertices and two vertices are linked by at most one edge. Two vertices linked by an edge are called *adjacent* or *neighbors*. The *neighborhood* of a vertex $v \in V$, denoted by $N(v)$, is the set of vertices adjacent to v. Two edges that share a common vertex are also called adjacent. A *loop* is an edge whose endpoints are equal. *Multiple edges* are edges having the same pair of endpoints. A simple graph has no loops nor multiple edges. In simple graphs, the edges can be named by the endpoints like an unordered set $\{v, v'\}$, where v and v' are vertices.

R. Portugal, *Quantum Walks and Search Algorithms*, Quantum Science and Technology, https://doi.org/10.1007/978-3-319-97813-0

The *degree* of vertex v is the number of edges incident to the vertex and is denoted by $d(v)$. The *maximum degree* is denoted by $\Delta(\Gamma)$, and the *minimum degree* is denoted by $\delta(\Gamma)$. A graph is *d-regular* if all vertices have degree d, that is, each vertex has exactly d neighbors. The *handshaking lemma* states that every graph has an even number of vertices with odd degree, which is a consequence of the *degree sum formula*

$$\sum_{v \in V} d(v) = 2|E|.$$

A *path* is a list $v_0, e_1, v_1, \ldots, e_k, v_k$ of vertices and edges such that edge e_i has endpoints v_{i-1} and v_i. A *cycle* is a closed path.

A graph is *connected* when there is a path between every pair of vertices; otherwise it is called *disconnected*. An example of connect graph is the *complete graph*, which denoted by K_N where N is the number of vertices, and is a simple graph in which every pair of distinct vertices is connected by an edge.

A subgraph $\Gamma'(V', E')$, where $V' \subset V$ and $E' \subset E$, is an *induced subgraph* of $\Gamma(V, E)$ if it has exactly the edges that appear in Γ over the same vertex set. If two vertices are adjacent in Γ, they are also adjacent in the induced subgraph. It is common to use the term subgraph in place of induced subgraph.

A graph Γ is *H-free* if Γ has no induced subgraph *isomorphic* to graph H. Take for instance a *diamond graph*, which is a graph with four vertices and five edges consisting of a K_4 minus one edge or two triangles sharing a common edge. A graph is *diamond-free* if no induced subgraph is isomorphic to a *diamond graph*.

The *adjacency matrix* M of a simple graph $\Gamma(V, E)$ is the symmetric square matrix whose rows and columns are indexed by the vertices and whose entries are

$$M_{vv'} = \begin{cases} 1, & \text{if } \{v, v'\} \in E(\Gamma), \\ 0, & \text{otherwise.} \end{cases}$$

The *Laplacian matrix* L of a simple graph $\Gamma(V, E)$ is the symmetric square matrix whose rows and columns are indexed by the vertices and whose entries are

$$L_{vv'} = \begin{cases} d(v), & \text{if } v = v', \\ -1, & \text{if } \{v, v'\} \in E(\Gamma), \\ 0, & \text{otherwise.} \end{cases}$$

Note that $L = D - A$, where D is the diagonal matrix whose rows and columns are indexed by the vertices and whose entries are $D_{vv'} = d(v)\delta_{vv'}$. The symmetric normalized Laplacian matrix is defined as $L^{\text{sym}} = D^{-1/2} L D^{-1/2}$.

Most of the times in this book, the term *graph* is used in place of *simple graph*. We also use the term *simple graph* to stress that the graph is undirected and has no loops nor multiple edges.

B.2 Multigraph

A *multigraph* is an extension of the definition of graph that allows multiple edges. Many books use the term graph as a synonym of multigraph. In a simple graph, the notation $\{v, v'\}$ is an edge label. In a multigraph, $\{v, v'\}$ does not characterize an edge and the edges can have their own identity or not. For quantum walks, we need to give labels for each edge (each one has its own identity). Formally, an *undirected labeled multigraph* $G(V, E, f)$ consists of a vertex set V, an edge *multiset* E, and an injective function $f : E \to \Sigma$, whose codomain Σ is an alphabet for the edge labels.

B.3 Bipartite Graph

A *bipartite graph* is a graph whose vertex set V is the union of two disjoint sets X and X' so that no two vertices in X are adjacent and no two vertices in X' are adjacent. A *complete bipartite graph* is a bipartite graph such that every possible edge that could connect vertices in X and X' is part of the graph and is denoted by $K_{m,n}$, where m and n are the cardinalities of sets X and X', respectively. $K_{m,n}$ is the graph that $V(K) = X \cup X'$ and $E(K) = \{\{x, x'\} : x \in X, x' \in X'\}$.

Theorem B.1. (König) *A graph is bipartite if and only if it has no odd cycle.*

B.4 Intersection Graph

Let $\{S_1, S_2, S_3, \ldots\}$ be a family of sets. The *intersection graph* of this family of sets is a graph whose vertices are the sets and two vertices are adjacent if and only if the intersection of the corresponding sets is nonempty, that is, $G(V, E)$ is the intersection graph of family $\{S_1, S_2, S_3, \ldots\}$ if $V = \{S_1, S_2, S_3, \ldots\}$ and $E(G) = \{\{S_i, S_j\} : S_i \cap S_j \neq \emptyset\}$ for all $i \neq j$.

B.5 Clique, Stable Set, and Matching

A *clique* is a subset of vertices of a graph such that its induced subgraph is complete. A *maximal clique* is a clique that cannot be extended by including one more adjacent vertex, that is, it is not contained in a larger clique. A *maximum clique* is a clique of maximum possible size. A clique of size d is called a *d-clique*. A set with one vertex is a clique. Some references in graph theory use the term "clique" as synonym of *maximal clique*. We avoid this notation here.

A *clique partition* of a graph Γ is a set of cliques of Γ that contains each edge of Γ exactly once. A *minimum clique partition* is a clique partition with the smallest set of cliques. A *clique cover* of a graph Γ is a set of cliques of Γ that contains each edge of Γ at least once. A *minimum clique cover* is a clique cover with the smallest set of cliques.

A *stable set* is a set of pairwise nonadjacent vertices.

A *matching* $M \subseteq E$ is a set of edges without pairwise common vertices. An edge $m \in M$ *matches* the endpoints of m. A *perfect matching* is a matching that matches all vertices of the graph.

B.6 Graph Operators

Let $\mathcal{C}$ be the set of all graphs. A graph operator $\mathcal{O} : \mathcal{C} \longrightarrow \mathcal{C}$ is a function that maps an arbitrary graph $G \in \mathcal{C}$ to another graph $G' \in \mathcal{C}$.

B.6.1 Clique Graph Operator

A *clique graph* $K(\Gamma)$ of a graph Γ is a graph such that every vertex represents a maximal clique of Γ and two vertices of $K(\Gamma)$ are adjacent if and only if the underlying maximal cliques in Γ share at least one vertex in common.

The clique graph of a *triangle-free graph* G is isomorphic to the line graph of G.

B.6.2 Line Graph Operator

A *line graph* (or *derived graph* or *interchange graph*) of a graph Γ (called *root graph*) is another graph $L(\Gamma)$ so that each vertex of $L(\Gamma)$ represents an edge of Γ and two vertices of $L(\Gamma)$ are adjacent if and only if their corresponding edges share a common vertex in Γ.

The line graph of a multigraph is a simple graph. On the other hand, given a simple graph G, it is possible to determine whether G is the line graph of a multigraph H, for instance, via the following theorems:

Theorem B.2. (Bermond and Meyer) *A simple graph G is a line graph of a multigraph if and only if there exists a family of cliques C in G such that*

1. *Every edge $\{v, v'\} \in E(G)$ belongs to at least one clique $c_i \in C$.*
2. *Every vertex $v \in V(G)$ belongs to exactly two cliques $c_i, c_j \in C$.*

A graph is *reduced* from a multigraph if the graph is obtained from a multigraph by merging multiple edges into single edges.

Theorem B.3. (Bermond and Meyer) *A simple graph is a line graph of a multigraph H if and only if the graph reduced from H is the line graph of a simple graph.*

It is possible to determine whether G is the line graph of a *bipartite multigraph* via the following theorem.

Theorem B.4. (Peterson) *A simple graph G is a line graph of a bipartite multigraph if and only if $K(G)$ is bipartite.*

B.6.3 Subdivision Graph Operator

A *subdivision* (or *expansion*) of a graph G is a new graph resulting from the subdivision of one or more edges in G. The *barycentric subdivision* subdivides all edges of the graph or a *multigraph* and produces a new bipartite simple graph. If the original graph is $G(V, E)$, the *barycentric subdivision* generates a new graph $BS(G) = \Gamma(V', E')$, whose vertex set is $V'(\Gamma) = V(G) \cup E(G)$ and an edge $\{v, e\}$, where $v \in V$ and $e \in E$, belongs to $E'(\Gamma)$ if and only if v is incident to e.

B.6.4 Clique–Insertion Operator

The *clique–insertion operator* replaces each vertex v of a graph G by a maximal $d(v)$-clique, creating a new graph $CI(G)$. Figure B.1 shows an example of a clique insertion, which replaces a vertex of degree 5 by a 5-clique. Note that the new clique is a maximal clique. Using the degree sum formula, the number of vertices of the clique–inserted graph is $|V(CI(G))| = 2|E(G)|$.

There is a relation between the clique–inserted graph and the line graph of the subdivision graph (called *para-line graph*).

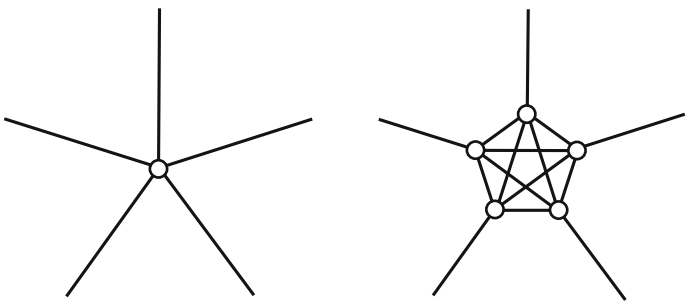


Fig. B.1 Example of a clique insertion. A degree-5 vertex (left-hand graph) is replaced by a 5-clique (right-hand graph)

Theorem B.5. (Sampathkumar) *The para-line graph of G is isomorphic to the clique–inserted graph $CI(G)$.*

B.7 Coloring

A *coloring* of a graph is a labeling of the vertices with colors so that no two vertices sharing the same edge have the same color. The smallest number of colors needed to color a graph Γ is called *chromatic number*, denoted by $\chi(\Gamma)$. A graph that can be assigned a coloring with k colors is *k-colorable* and is *k-chromatic* if its chromatic number is exactly k.

Theorem B.6. (Brooks) $\chi(\Gamma) \leq \Delta(\Gamma)$ *for a graph* Γ, *unless* Γ *is a complete graph or an odd cycle.*

The complete graph with N vertices has $\chi(\Gamma) = N$ and $\Delta(\Gamma) = N - 1$. Odd cycles have $\chi(\Gamma) = 3$ and $\Delta(\Gamma) = 2$. For these graphs the bound $\chi(\Gamma) \leq \Delta(\Gamma) + 1$ is the best possible. In all other cases, the bound $\chi(\Gamma) \leq \Delta(\Gamma)$ is given by Brooks' theorem.

The concept of coloring can be applied to the edge set of a loop free graph. An *edge coloring* is a coloring of the edges so that no vertex is incident to two edges of the same color. The smallest number of colors needed for an edge coloring is called the *chromatic index* or *edge-chromatic number*, denoted by $\chi'(\Gamma)$.

Theorem B.7. (Vizing) *A graph* Γ *of maximal degree* $\Delta(\Gamma)$ *has edge-chromatic number* $\Delta(\Gamma)$ *or* $\Delta(\Gamma) + 1$, *that is,* $\Delta(\Gamma) \leq \chi'(\Gamma) \leq \Delta(\Gamma) + 1$.

Since at least $\Delta(\Gamma)$ colors are always necessary for edge coloring, the set of all graphs may be partitioned into two classes: (1) *class* 1 graphs for which $\Delta(\Gamma)$ colors are sufficient and (2) *class* 2 graphs for which $\Delta(\Gamma) + 1$ colors are necessary. Examples of graphs in class 1 are: complete graphs K_N for even N, bipartite graphs. Examples of graph in class 2 are: regular graphs with an odd number of vertices $N > 1$ (includes complete graphs K_N for odd $N \geq 3$), Petersen graph. To determine whether an arbitrary graph is in class 1 is *NP-complete*. There are asymptotic results in literature showing that the proportion of graphs in class 2 is very small.

Given a graph Γ in class 2, we describe two ways to modify the graph in order to create a new graph in class 1: (1) Add a *leaf* to each vertex of Γ, or (2) make an identical copy of Γ and add edges connecting the pairs of identical vertices.

B.8 Diameter

The *geodesic distance* (simply *distance*) between two vertices in graph $G(V, E)$ is the number of edges in a shortest path connecting them. The *eccentricity* $\epsilon(v)$ of a

vertex v is the greatest geodesic distance between v and any other vertex, that is, it is how far a vertex is from the vertex most distant from it in the graph. The *diameter* d of a graph is $d = \max_{v \in V} \epsilon(v)$, that is, it is the maximum eccentricity of any vertex in the graph or the greatest distance between any pair of vertices.

B.9 Directed Graph

A *directed graph* or *digraph* G is defined by a vertex set $V(G)$, an arc set $A(G)$, and a function assigning each arc an ordered pair of vertices. We use the notation (v, v') for an ordered pair of vertices, where v is the *tail* and v' is the *head*, and (v, v') is called *directed edge* or simply *arc*. A digraph is a *simple digraph* if each ordered pair is the head and tail of at most one arc. The *underlying graph* of a digraph G is the graph obtained by considering the arcs of G as unordered pairs.

If (v, v') and (v', v) are in $A(G)$, the set with (v, v') and (v', v) is called a *pair of symmetric arcs*. A *symmetric directed graph* G or *symmetric digraph* is a digraph whose arc set comprises pairs of symmetric arcs, that is, if $(v, v') \in A(G)$, then $(v', v) \in A(G)$. Figure B.2 depicts an example of a symmetric digraph G and its underlying simple graph H.

The *outdegree* $d^+(v)$ is the number of arcs with tail v. The *indegree* $d^-(v)$ is the number of arcs with head v. The definitions of *out-neighborhood*, *in-neighborhood*, *minimum* and *maximum indegree* and *outdegree* are straightforward generalizations of the corresponding undirected ones. A *local sink* or simply *sink* is a vertex with outdegree zero, and a *local source* or simply *source* is a vertex with indegree zero. A *global sink* is a vertex which is reached by all other vertices. A *global source* A is a vertex which reaches all other vertices.

A *directed cycle graph* is a directed version of a cycle graph, where all edges are oriented in the same direction. A *directed acyclic graph* is a finite directed graph with no directed cycles. The *moral graph* of a directed acyclic graph G is a simple graph that is obtained from the underlying simple graph of G by adding edges between all pairs of vertices that have a common child (in G).

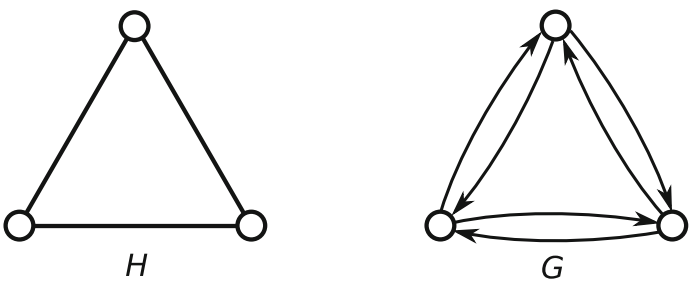


Fig. B.2 Example of a symmetric digraph G and its underlying simple graph H

B.10 Some Named Graphs

B.10.1 Johnson Graphs

Let $[N]$ be the set $\{1, \ldots, N\}$. There are $\binom{N}{k}$ k-subsets of $[N]$, where a k-subset is a subset of $[N]$ with k elements. Let us define the Johnson graph $J(N, k)$. The vertices of $J(N, k)$ are the k-subsets of $[N]$, and two vertices are adjacent if and only if their intersection has size $(k - 1)$. If $k = 1$, $J(N, 1)$ is the complete graph K_N. $J(N, k)$ and $J(N, N - k)$ are the same graphs after renaming the vertices. $J(N, k)$ is a regular graph with degree $k\,(N - k)$. The diameter of $J(N, k)$ is $\min(k, N - k)$.

B.10.2 Kneser Graphs

Let $[N]$ be the set $\{1, \ldots, N\}$. A k-subset is a subset of $[N]$ with k elements. The *Kneser graph* $KG_{N,k}$ is the graph whose vertices are the k-subsets, and two vertices are adjacent if and only if the two corresponding sets are disjoint. If $k = 1$, $KG_{N,1}$ is the complete graph K_N. $KG_{N,k}$ is a regular graph with degree $\binom{N-k}{k}$. The diameter of $KG_{N,k}$ is $\lceil (k-1)/(N-2k) \rceil + 1$. The *Petersen graph*, depicted in Fig. B.3, is a Kneser graph $KG_{5,2}$. It is in class 2 because it is 3-regular and its edge-chromatic number is 4.

B.10.3 Cayley Graphs

A *Cayley graph* $\Gamma(G, S)$ encodes the structure of a *group* G described by a *generating set* S in the context of *abstract algebra*.

Definition B.8. A *group* is a nonempty set G together with a binary operation $\cdot$ (called product), which satisfies the following requirements:

- (Closure) For all a, b in G, $a \cdot b$ is also in G.
- (Associativity) For all a, b, c in G, $(a \cdot b) \cdot c = a \cdot (b \cdot c)$.

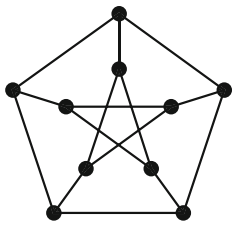


Fig. B.3 Petersen graph

- (Identity) There exists an identity element e in G such that, for every element a in G, $a \cdot e = e \cdot a = a$.
- (Inverse) For each a in G, there exists an element b in G, such that $a \cdot b = b \cdot a = e$, where e is the identity element. Element b is denoted by a^{-1}.

The *order* of a group is its number of elements. A group is finite if its order is finite. A group is *commutative* or *abelian* if the binary operation is commutative. A *generating set* of a group G is a subset $S \subset G$ such that every element of G can be expressed as the product of finitely many elements of S and their inverses. From now on, we suppose that S is finite. S is called *symmetric* if $S = S^{-1}$, that is, whenever $s \in S$, s^{-1} is also in S.

A *subgroup* of a group G is a subset H of G such that H is a group with the same product operation of G. No proper subgroup of group G can contain a generating set of G.

The Cayley graph $\Gamma(G, S)$ is a directed graph defined as follows. The vertex set $V(\Gamma)$ is G, and the arc (a, b) is in $A(\Gamma)$ if and only if $b = a \cdot s$ for some $s \in S$, where $a, b \in G$.

If S is symmetric and $e \neq S$, the Cayley graph $\Gamma(G, S)$ is a $|S|$-regular simple graph. It is a difficult problem to decide whether a Cayley graph of a group described by a symmetric generating set is in class 1 or class 2. There is a remarkable conjecture studied over decades:

Conjecture B.9. (Stong) *All undirected Cayley graphs of groups of even order are in class 1.*

Further Reading

Graph theory has many applications, and it is easy to get lost and waste time after taking some wrong direction. No danger comes from those introductory books [53, 139, 314, 326]. Before starting to read an advanced book, check whether it is really necessary to go further. In the context of quantum walks, the survey [58] is useful. Harary's book [138] is excellent (there is a new edition by CRC Press). Other suggestions are [75, 102, 121]. Wikipedia (English version) is an excellent place to obtain quickly the definition or the main properties of a concept in graph theory, and `http://www.graphclasses.org` is a Web site used by researchers in graph theory. Some results compiled in this Appendix are described in papers [259, 289, 304, 318, 347].

Appendix C
Classical Hitting Time

Consider a connected, nondirected, and non-bipartite *graph* $\Gamma(X, E)$, where $X = \{x_1, \ldots, x_n\}$ is the vertex set and E is the edge set. The *hitting time* of a *classical random walk* on this graph is the *expected time* for the walker to reach a marked vertex for the first time, once given the initial conditions. We may have more than one marked vertex defined by a subset $M \in X$. In this case, the hitting time is the expected time for the walker to reach a vertex in M for the first time.

If $p_{x\,x'}(t)$ is the probability of the walker to reach x' for the first time at time t having left x at $t = 0$, the hitting time from vertex x to x' is

$$H_{x\,x'} = \sum_{t=0}^{\infty} t\; p_{x\,x'}(t). \tag{C.1}$$

Define $H_{x\,x} = 0$ when the departure and arrival vertices are the same.

For example, the probability $p_{x\,x'}(t)$ at time $t = 1$ when $x \neq x'$ for the *complete graph* with n vertices is $1/(n-1)$, because the walker has $n-1$ possible vertices to move in the first step. To arrive at vertex x' at time $t = 2$ for the first time, the walker must visit one of $n-2$ vertices different from x and x'. The probability is $(n-2)/(n-1)$. After this visit, it must go directly to vertex x', which occurs with probability $1/(n-1)$. Therefore, $p_{x\,x'}(2) = (n-2)/(n-1)^2$. Generalizing this argumentation, we obtain $p_{x\,x'}(t) = (n-2)^{t-1}/(n-1)^t$. Then,

$$H_{x\,x'} = \sum_{t=0}^{\infty} t\,\frac{(n-2)^{t-1}}{(n-1)^t}.$$

Using the identity $\sum_{t=0}^{\infty} t\alpha^t = \alpha/(1-\alpha)^2$, which is valid for $0 < \alpha < 1$, we obtain

$$H_{x\,x'} = n - 1. \tag{C.2}$$

Usually, the hitting time depends on x and x', but the complete graph is an exception. In the general case, $H_{x\,x'}$ can be different from $H_{x'x}$.

R. Portugal, *Quantum Walks and Search Algorithms*, Quantum Science and Technology, https://doi.org/10.1007/978-3-319-97813-0

The notion of hitting time from a vertex to a subset can be formalized as follows: Suppose that M is a nonempty subset of X with cardinality m and define $p_{xM}(t)$ as the probability that the walker reaches any of the vertices in M for the first time at time t having left x at $t = 0$. The hitting time from x to M is

$$H_{xM} = \sum_{t=0}^{\infty} t\, p_{xM}(t). \tag{C.3}$$

Again, we define $H_{xM} = 0$ if $x \in M$.

Let us use an extended notion of hitting time when the walker starts from a probability distribution. In the former case, the probability to depart from vertex x is 1 and the probability to depart from any other vertex is 0. Suppose that the walker starts with a distribution σ, that is, at the initial time the probability of the walker to be at vertex x is σ_x. The most used initial distributions are the *uniform distribution* $\sigma_x = 1/n$ and the *stationary distribution*, which is defined ahead. In any case, the initial distribution must satisfy $\sum_{x\in X} \sigma_x = 1$. The hitting time from σ to M is

$$H_{\sigma M} = \sum_{x\in X} \sigma_x\, H_{xM}. \tag{C.4}$$

That is, $H_{\sigma M}$ is the *expected value* of the hitting time H_{xM} from x to M weighted with distribution σ.

Exercise C.1. Show that for the complete graph

$$H_{xM} = \frac{n-1}{m}$$

if $x \notin M$.

Exercise C.2. Show that for the complete graph

$$H_{\sigma M} = \frac{(n-m)(n-1)}{mn}$$

if σ is the uniform distribution. Why $H_{\sigma M} \approx H_{xM}$ for $n \gg m$?

C.1 Hitting Time Using the Stationary Distribution

Equations (C.1) and (C.3) are troublesome for the practical calculation of the hitting time of random walks on graphs. Fortunately, there are alternative methods. The best-known method uses a recursive method. Let us illustrate this method using the complete graph. We want to calculate $H_{x\,x'}$. The walker departs from x and moves directly to x' with probability $1/(n-1)$ spending one time unit. With probability

$(n-2)/(n-1)$, the walker moves to vertex x'' different from x' and therefore it spends one time unit plus the expected time to go from x'' to x', which is $H_{x\,x'}$. We have established the following recursive equation:

$$H_{x\,x'} = \frac{1}{n-1} + \frac{n-2}{n-1}\left(1 + H_{x\,x'}\right), \tag{C.5}$$

the solution of which is equal to (C.2).

This method works for an arbitrary graph. If V_x is the *neighborhood* of x, the cardinality of V_x is the *degree* of x denoted by $d(x)$. To help this calculation, we assume that the distance between x and x' is greater than 1. So, the walker will depart from x and will move to the neighboring vertex x'' with probability $1/d(x)$ spending one time unit. Now, we must add this result to the expected time to move from x'' to x'. This has to be performed for all vertices x'' in the neighborhood of x. We obtain

$$H_{x\,x'} = \frac{1}{d(x)} \sum_{x'' \in V_x} \left(1 + H_{x''\,x'}\right). \tag{C.6}$$

Equation (C.5) is a special case of (C.6), because for the complete graph $d(x) = n-1$ and $H_{x''\,x'} = H_{x\,x'}$ unless $x'' = x'$. The case $x'' = x'$ generates the first term in (C.5). The remaining $n-2$ cases generate the second term. This shows that (C.6) is general and the distance between x and x' need not be greater than 1. However, we cannot take $x = x'$ (distance 0) since the left-hand side is zero and the right-hand side is not.

The goal now is to solve (C.6) in terms of the hitting time $H_{x\,x'}$. This task is facilitated if (C.6) is converted to the matrix form. If H is a square n-dimensional matrix with entries $H_{x\,x'}$, the left-hand side will be converted into H and the right-hand side must be expanded. Using that

$$p_{x\,x'} = \begin{cases} \frac{1}{d(x)}, & \text{if } x' \text{ is adjacent to } x; \\ 0, & \text{otherwise,} \end{cases} \tag{C.7}$$

we obtain the following matrix equation:

$$H = J + PH + D, \tag{C.8}$$

where J is a matrix with all entries equal to 1, P is the *right stochastic matrix*, and D is a diagonal matrix that should be introduced to validate the matrix equation for the diagonal elements. P is also called *transition matrix* or *probability matrix*, as we have discussed in Chap. 3.

The diagonal matrix D can be calculated using the *stationary distribution* π, which is the distribution that satisfies equation $\pi^T \cdot P = \pi^T$. It is also called *limiting* or *equilibrium distribution*. For connected, nondirected, and non-bipartite graphs $\Gamma(X, E)$, there is always a limiting distribution. By left multiplying (C.8) by π^T, we obtain

$$D_{x\,x} = -\frac{1}{\pi_x},$$

where π_x is the xth entry of π.

Equation (C.8) can be written as $(I - P)H = J + D$. When we try to find H using this equation, we deal with the fact that $(I - P)$ is a noninvertible matrix, because $\mathbf{1}$ is a 0-eigenvector of $(I - P)$, where $\mathbf{1}$ is the vector with all entries equal to 1. This means that equation $(I - P)X = J + D$ has more than one solution X. In fact, if matrix X is a solution, then $X + \mathbf{1} \cdot v^T$ is also a solution for any vector v. However, having at hand a solution X of this equation does not guarantee that we have found H. There is a way to verify whether X is a correct solution by using that $H_{x\,x}$ must be zero for all x. A solution of equation $(I - P)X = J + D$ is

$$X = \left(I - P + \mathbf{1} \cdot \pi^T\right)^{-1} (J + D), \tag{C.9}$$

as can be checked by solving Exercise C.3. Now we add a term of type $\mathbf{1} \cdot v^T$ to cancel out the diagonal entries of X, and we obtain

$$H = X - \mathbf{1} \cdot v^T, \tag{C.10}$$

where the entries of vector v are the diagonal entries of X, that is, $v_x = X_{x\,x}$.

Exercise C.3. Let

$$M = I - P + \mathbf{1} \cdot \pi^T.$$

1. Show that M is invertible.
2. Using equations $\pi^T \cdot P = \pi^T$, $P \cdot \mathbf{1} = \mathbf{1}$, and

$$M^{-1} = \sum_{t=0}^{\infty} (I - M)^t,$$

show that

$$M^{-1} = \mathbf{1} \cdot \pi^T + \sum_{t=0}^{\infty} \left(P^t - \mathbf{1} \cdot \pi^T\right).$$

3. Show that solution (C.9) satisfies equation $(I - P)X = J + D$.
4. Show that matrix H given by (C.10) satisfies $H_{x\,x} = 0$.

Exercise C.4. Find the stochastic matrix of the complete graph with n vertices. Using the fact that the stationary distribution is uniform in this graph, find matrix X using (C.9) and then find matrix H using (C.10). Check the results with (C.2).

C.2 Hitting Time Without the Stationary Distribution

There is an alternative method for calculating the hitting time that does not use the stationary distribution. We describe the method using $H_{\sigma M}$ as defined in (C.4). The vertices in M are called *marked vertices*. Consider the *symmetric digraph* whose underlying graph is $\Gamma(X, E)$. Now we define a *modified digraph*, which is obtained from the symmetric digraph by converting all *arcs* leaving the marked vertices into *loops*, while maintaining unchanged the incoming ones. This means that if the walker reaches a marked vertex, the walker will stay there forever. To calculate the hitting time, the original undirected graph and the modified digraph are equivalent. However, the stochastic matrices are different. Let us denote the stochastic matrix of the modified graph by P', whose entries are

$$p'_{xy} = \begin{cases} p_{xy}, & x \notin M; \\ \delta_{xy}, & x \in M. \end{cases} \tag{C.11}$$

What is the probability of finding the walker in $X \backslash M$ at time t before visiting M? Let $\sigma^{(0)}$ be the initial probability distribution on the vertices of the original graph viewed as a row vector. Then, the distribution after t steps is

$$\sigma^{(t)} = \sigma^{(0)} \cdot P^t. \tag{C.12}$$

Let $\mathbf{1}$ be the column n-vector with all entries equal to 1. Define $\mathbf{1}_{X \backslash M}$ as the column n-vector with $n - m$ entries equal to 1 corresponding to the vertices that are in $X \backslash M$ and m entries equal to zero corresponding to the vertices are in M. The probability of finding the walker in $X \backslash M$ at time t is $\sigma^{(t)} \cdot \mathbf{1}_{X \backslash M}$. However, this expression is not useful for calculating the hitting time, because the walker has already visited M. We want to find the probability of the walker being in $X \backslash M$ at time t having not visited M. This result is obtained if we use matrix P' instead of P in (C.12). In fact, if the evolution is driven by matrix P' and the walker has visited M, it remains imprisoned in M forever. Therefore, if the walker is found in $X \backslash M$, it has certainly not visited M. The probability of finding the walker in $X \backslash M$ at time t without having visited M is $\sigma^{(0)} \cdot (P')^t \cdot \mathbf{1}_{X \backslash M}$.

In (C.3), we have calculated the average time to reach a marked vertex for the first time employing the usual formula for calculating weighted averages. When the variable t assumes nonnegative integer values, there is an alternative formula for calculating this average. This formula applies to this context because time t is the number of steps. Let T be the number of steps to reach a marked vertex for the first time, and let $p(T \geq t)$ be the probability of reaching M for the first time for any number of steps T equal to or greater than t. If the initial condition is distribution σ, the hitting time can be equivalently defined by formula

$$H_{\sigma M} = \sum_{t=1}^{\infty} p(T \geq t). \tag{C.13}$$

To verify the equivalence of this new formula with the previous one, note that

$$p(T \geq t) = \sum_{j=t}^{\infty} p(T = j), \tag{C.14}$$

where $p(T = t)$ is the probability of reaching M for the first time with exactly t steps. Using (C.14) and (C.13), we obtain

$$\begin{aligned} H_{\sigma M} &= \sum_{j=1}^{\infty} \sum_{t=1}^{j} p(T = j) \\ &= \sum_{j=1}^{\infty} j\, p(T = j). \end{aligned} \tag{C.15}$$

This last equation is equivalent to (C.3).

We can give another interpretation for probability $p(T \geq t)$. If the walker reaches M at $T \geq t$, then in the first $t - 1$ steps it will still be in $X \backslash M$, that is, it will be on one of the unmarked vertices without having visited M. We have learned in a previous paragraph that the probability of the walker being in $X \backslash M$ at time t without having visited M is $\sigma^{(0)} \cdot (P')^{t-1} \cdot \mathbf{1}_{X \backslash M}$. Then,

$$p(T \geq t) = \sigma^{(0)} \cdot (P')^{t-1} \cdot \mathbf{1}_{X \backslash M}. \tag{C.16}$$

Define $P_{\overline{M}}$ as a square $(n - m)$-matrix obtained from P by deleting the rows and columns corresponding to vertices of M. Define $\sigma_{\overline{M}}$ and $\mathbf{1}_{\overline{M}}$ using the same procedure. Analyzing the entries that do not vanish after multiplying the matrices on the right-hand side of (C.16), we conclude that

$$p(T \geq t) = \sigma_{\overline{M}}^{(0)} \cdot P_{\overline{M}}^{t-1} \cdot \mathbf{1}_{\overline{M}}. \tag{C.17}$$

Using the above equation and (C.13), we obtain

$$\begin{aligned} H_{\sigma M} &= \sigma_{\overline{M}}^{(0)} \cdot \left(\sum_{t=0}^{\infty} P_{\overline{M}}^{t} \right) \cdot \mathbf{1}_{\overline{M}} \\ &= \sigma_{\overline{M}}^{(0)} \cdot \left(I - P_{\overline{M}} \right)^{-1} \cdot \mathbf{1}_{\overline{M}}. \end{aligned} \tag{C.18}$$

Matrix $(I - P_{\overline{M}})$ is always invertible for connected, nondirected, and non-bipartite graphs. This result follows from the fact that $\mathbf{1}$ is not an eigenvector of $P_{\overline{M}}$, and hence $(I - P_{\overline{M}})$ has no eigenvalue equal to 0.

The strategy used to obtain (C.18) is used to define the *quantum hitting time* in *Szegedy's model*.

Exercise C.5. Use (C.18) to find the hitting time of a random walk on the complete graph with n vertices, and compare the results with Exercises C.1 and C.2.

Further Reading

The classical hitting time is described in many references, for instance, [11, 215, 235, 245]. The last chapter of [235] describes in detail the *Perron–Frobenius theorem*, which is important in the context of this appendix.

References

1. Aaronson, S., Ambainis, A.: Quantum search of spatial regions. In: Theory of Computing, pp. 200–209 (2003)
2. Aaronson, S., Shi, Y.: Quantum lower bounds for the collision and the element distinctness problems. J. ACM **51**(4), 595–605 (2004)
3. Abal, G., Donangelo, R., Forets, M., Portugal, R.: Spatial quantum search in a triangular network. Math. Struct. Comput. Sci. **22**(03), 521–531 (2012)
4. Abal, G., Donangelo, R., Marquezino, F.L., Portugal, R.: Spatial search on a honeycomb network. Math. Struct. Comput. Sci. **20**(6), 999–1009 (2010)
5. Abreu, A., Cunha, L., Fernandes, T., de Figueiredo, C., Kowada, L., Marquezino, F., Posner, D., Portugal, R.: The graph tessellation cover number: extremal bounds, efficient algorithms and hardness. In: Bender, M.A., Farach-Colton, M., Mosteiro, M.A. (eds.) LATIN 2018. Lecture Notes in Computer Science, vol. 10807, pp. 1–13. Springer, Berlin (2018)
6. Agliari, E., Blumen, A., Mülken, O.: Quantum-walk approach to searching on fractal structures. Phys. Rev. A **82**, 012305 (2010)
7. Aharonov, D.: Quantum computation - a review. In: Stauffer, D. (ed.) Annual Review of Computational Physics, vol. VI, pp. 1–77. World Scientific, Singapore (1998)
8. Aharonov, D., Ambainis, A., Kempe, J., Vazirani, U.: Quantum walks on graphs. In: Proceedings of the 33th STOC, pp. 50–59. ACM, New York (2001)
9. Aharonov, Y., Davidovich, L., Zagury, N.: Quantum random walks. Phys. Rev. A **48**(2), 1687–1690 (1993)
10. Alberti, A., Alt, W., Werner, R., Meschede, D.: Decoherence models for discrete-time quantum walks and their application to neutral atom experiments. New J. Phys. **16**(12), 123052 (2014)
11. Aldous, D., Fill, J.A.: Reversible Markov chains and random walks on graphs (2002). http://www.stat.berkeley.edu/~aldous/RWG/book.html
12. Alvir, R., Dever, S., Lovitz, B., Myer, J., Tamon, C., Xu, Y., Zhan, H.: Perfect state transfer in laplacian quantum walk. J. Algebr. Comb. **43**(4), 801–826 (2016)
13. Ambainis, A.: Quantum walks and their algorithmic applications. Int. J. Quantum Inf. **01**(04), 507–518 (2003)
14. Ambainis, A.: Quantum walk algorithm for element distinctness. In: Proceedings 45th Annual IEEE Symposium on Foundations of Computer Science FOCS, pp. 22–31. Washington (2004)
15. Ambainis, A.: Polynomial degree and lower bounds in quantum complexity: collision and element distinctness with small range. Theory Comput. **1**, 37–46 (2005)
16. Ambainis, A.: Quantum walk algorithm for element distinctness. SIAM J. Comput. **37**(1), 210–239 (2007)

R. Portugal, *Quantum Walks and Search Algorithms*, Quantum Science and Technology, https://doi.org/10.1007/978-3-319-97813-0

17. Ambainis, A., Bach, E., Nayak, A., Vishwanath, A., Watrous, J.: One-dimensional quantum walks. In: Proceedings of the 33th STOC, pp. 60–69. ACM, New York (2001)
18. Ambainis, A., Bačkurs, A., Nahimovs, N., Ozols, R., Rivosh, A.: Search by quantum walks on two-dimensional grid without amplitude amplification. In: Proceedings of the 7th TQC, Tokyo, Japan, pp. 87–97. Springer, Berlin (2013)
19. Ambainis, A., Kempe, J., Rivosh, A.: Coins make quantum walks faster. In: Proceedings of the 16th Annual ACM-SIAM Symposium on Discrete Algorithms SODA, pp. 1099–1108 (2005)
20. Ambainis, A., Portugal, R., Nahimov, N.: Spatial search on grids with minimum memory. Quantum Inf. Comput. **15**, 1233–1247 (2015)
21. Ampadu, C.: Return probability of the open quantum random walk with time-dependence. Commun. Theor. Phys. **59**(5), 563 (2013)
22. Anderson, P.W.: Absence of diffusion in certain random lattices. Phys. Rev. **109**, 1492–1505 (1958)
23. Angeles-Canul, R.J., Norton, R.M., Opperman, M.C., Paribello, C.C., Russell, M.C., Tamon, C.: Quantum perfect state transfer on weighted join graphs. Int. J. Quantum Inf. **7**(8), 1429–1445 (2009)
24. Apostol, T.M.: Calculus, Volume 1: One-Variable Calculus with an Introduction to Linear Algebra. Wiley, New York (1967)
25. Arunachalam, S., De Wolf, R.: Optimizing the number of gates in quantum search. Quantum Inf. Comput. **17**(3–4), 251–261 (2017)
26. Asbóth, J.K.: Symmetries, topological phases, and bound states in the one-dimensional quantum walk. Phys. Rev. B **86**, 195414 (2012)
27. Asbóth, J.K., Edge, J.M.: Edge-state-enhanced transport in a two-dimensional quantum walk. Phys. Rev. A **91**, 022324 (2015)
28. Axler, S.: Linear Algebra Done Right. Springer, New York (1997)
29. Balu, R., Liu, C., Venegas-Andraca, S.E.: Probability distributions for Markov chain based quantum walks. J. Phys. A: Math. Theor. **51**(3), 035301 (2018)
30. Barnett, S.: Quantum Information. Oxford University Press, New York (2009)
31. Barr, K., Fleming, T., Kendon, V.: Simulation methods for quantum walks on graphs applied to formal language recognition. Nat. Comput. **14**(1), 145–156 (2015)
32. Barr, K.E., Proctor, T.J., Allen, D., Kendon, V.M.: Periodicity and perfect state transfer in quantum walks on variants of cycles. Quantum Inf. Comput. **14**(5–6), 417–438 (2014)
33. Bašić, M.: Characterization of quantum circulant networks having perfect state transfer. Quantum Inf. Process. **12**(1), 345–364 (2013)
34. Beame, P., Saks, M., Sun, X., Vee, E.: Time-space trade-off lower bounds for randomized computation of decision problems. J. ACM **50**(2), 154–195 (2003)
35. Bednarska, M., Grudka, A., Kurzynski, P., Luczak, T., Wójcik, A.: Quantum walks on cycles. Phys. Lett. A **317**(1–2), 21–25 (2003)
36. Bednarska, M., Grudka, A., Kurzynski, P., Luczak, T., Wójcik, A.: Examples of non-uniform limiting distributions for the quantum walk on even cycles. Int. J. Quantum Inf. **2**(4), 453–459 (2004)
37. Belovs, A.: Learning-graph-based quantum algorithm for k-distinctness. In: IEEE 53rd Annual Symposium on Foundations of Computer Science, pp. 207–216 (2012)
38. Belovs, A., Childs, A.M., Jeffery, S., Kothari, R., Magniez, F.: Time-efficient quantum walks for 3-distinctness. In: Proceedings of the 40th International Colloquium ICALP, Riga, Latvia, 2013, pp. 105–122. Springer, Berlin (2013)
39. Benedetti, C., Buscemi, F., Bordone, P., Paris, M.G.A.: Non-Markovian continuous-time quantum walks on lattices with dynamical noise. Phys. Rev. A **93**, 042313 (2016)
40. Benenti, G., Casati, G., Strini, G.: Principles of Quantum Computation And Information: Basic Tools And Special Topics. World Scientific Publishing, River Edge (2007)
41. Benioff, P.: Space Searches with a Quantum Robot. AMS Contemporary Mathematics Series, vol. 305. American Mathematical Society, Providence (2002)

42. Bennett, C.H., Bernstein, E., Brassard, G., Vazirani, U.V.: Strengths and weaknesses of quantum computing. SIAM J. Comput. **26**(5), 1510–1523 (1997)
43. Bergou, J.A., Hillery, M.: Introduction to the Theory of Quantum Information Processing. Springer, New York (2013)
44. Bernard, P.-A., Chan, A., Loranger, É., Tamon, C., Vinet, L.: A graph with fractional revival. Phys. Lett. A **382**(5), 259–264 (2018)
45. Bernasconi, A., Godsil, C., Severini, S.: Quantum networks on cubelike graphs. Phys. Rev. A **78**, 052320 (2008)
46. Berry, S.D., Bourke, P., Wang, J.B.: QwViz: visualisation of quantum walks on graphs. Comput. Phys. Commun. **182**(10), 2295–2302 (2011)
47. Bhattacharya, N., van Linden van den Heuvell, H.B., Spreeuw, R.J.C., : Implementation of quantum search algorithm using classical Fourier optics. Phys. Rev. Lett. **88**, 137901 (2002)
48. Bian, Z.-H., Li, J., Zhan, X., Twamley, J., Xue, P.: Experimental implementation of a quantum walk on a circle with single photons. Phys. Rev. A **95**, 052338 (2017)
49. Biham, O., Nielsen, M.A., Osborne, T.J.: Entanglement monotone derived from Grover's algorithm. Phys. Rev. A **65**, 062312 (2002)
50. Boada, O., Novo, L., Sciarrino, F., Omar, Y.: Quantum walks in synthetic gauge fields with three-dimensional integrated photonics. Phys. Rev. A **95**, 013830 (2017)
51. Boettcher, S., Falkner, S., Portugal, R.: Renormalization group for quantum walks. J. Phys. Conf. Ser. **473**(1), 012018 (2013)
52. Boettcher, S., Li, S.: Analysis of coined quantum walks with renormalization. Phys. Rev. A **97**, 012309 (2018)
53. Bondy, A., Murty, U.S.R.: Graph Theory. Graduate Texts in Mathematics. Springer, London (2011)
54. Bose, S.: Quantum communication through an unmodulated spin chain. Phys. Rev. Lett. **91**, 207901 (2003)
55. Botsinis, P., Babar, Z., Alanis, D., Chandra, D., Nguyen, H., Ng, S.X., Hanzo, L.: Quantum error correction protects quantum search algorithms against decoherence. Sci. Rep. **6**, 38095 (2016)
56. Bougroura, H., Aissaoui, H., Chancellor, N., Kendon, V.: Quantum-walk transport properties on graphene structures. Phys. Rev. A **94**, 062331 (2016)
57. Boyer, M., Brassard, G., Høyer, P., Tapp, A.: Tight bounds on quantum searching. Forstschritte Der Physik **4**, 820–831 (1998)
58. Brandstädt, A., Le, V.B., Spinrad, J.P.: Graph Classes: A Survey. SIAM, Philadelphia (1999)
59. Brassard, G., Høyer, P., Mosca, M., Tapp, A.: Quantum amplitude amplification and estimation. Quantum Computation and Quantum Information Science. AMS Contemporary Mathematics Series, vol. 305, pp. 53–74. American Mathematical Society, Providence (2002)
60. Brassard, G., Høyer, P., Tapp, A.: Quantum cryptanalysis of hash and claw-free functions. In: Proceedings of the 3rd Latin American Symposium LATIN'98, pp. 163–169. Springer, Berlin (1998)
61. Breuer, H.P., Petruccione, F.: The Theory of Open Quantum Systems. Oxford University Press, Oxford (2002)
62. Bru, L.A., de Valcárcel, G.J., di Molfetta, G., Pérez, A., Roldán, E., Silva, F.: Quantum walk on a cylinder. Phys. Rev. A **94**, 032328 (2016)
63. Bruderer, M., Plenio, M.B.: Decoherence-enhanced performance of quantum walks applied to graph isomorphism testing. Phys. Rev. A **94**, 062317 (2016)
64. Buhrman, H., Dürr, C., Heiligman, M., Høyer, P., Magniez, F., Santha, M., de Wolf, R.: Quantum algorithms for element distinctness. SIAM J. Comput. **34**(6), 1324–1330 (2005)
65. Byrnes, T., Forster, G., Tessler, L.: Generalized Grover's algorithm for multiple phase inversion states. Phys. Rev. Lett. **120**, 060501 (2018)
66. Cáceres, M.O.: On the quantum CTRW approach. Eur. Phys. J. B **90**(4), 74 (2017)
67. Cameron, S., Fehrenbach, S., Granger, L., Hennigh, O., Shrestha, S., Tamon, C.: Universal state transfer on graphs. Linear Algebra Appl. **455**, 115–142 (2014)

68. Carteret, H.A., Ismail, M.E.H., Richmond, B.: Three routes to the exact asymptotics for the one-dimensional quantum walk. J. Phys. A: Math. Gen. **36**(33), 8775–8795 (2003)
69. Carvalho, S.L., Guidi, L.F., Lardizabal, C.F.: Site recurrence of open and unitary quantum walks on the line. Quantum Inf. Process. **16**(1), 17 (2016)
70. Cedzich, C., Geib, T., Grünbaum, F.A., Stahl, C., Velázquez, L., Werner, A.H., Werner, R.F.: The topological classification of one-dimensional symmetric quantum walks. Ann. Henri Poincaré **19**(2), 325–383 (2018)
71. Chakraborty, K., Maitra, S.: Application of Grover's algorithm to check non-resiliency of a boolean function. Cryptogr. Commun. **8**(3), 401–413 (2016)
72. Chakraborty, S., Novo, L., Di Giorgio, S., Omar, Y.: Optimal quantum spatial search on random temporal networks. Phys. Rev. Lett. **119**, 220503 (2017)
73. Chan, A., Coutinho, G., Tamon, C., Vinet, L., Zhan, H.: Quantum fractional revival on graphs (2018). arXiv:1801.09654
74. Chandrashekar, C.M., Busch, T.: Quantum percolation and transition point of a directed discrete-time quantum walk. Sci. Rep. **4**, 6583 (2014)
75. Chartrand, G., Lesniak, L., Zhang, P.: Graphs & Digraphs. Chapman & Hall/CRC, Boca Raton (2010)
76. Chiang, C.-F., Gomez, G.: Hitting time of quantum walks with perturbation. Quantum Inf. Process. **12**(1), 217–228 (2013)
77. Chiang, C.-F., Nagaj, D., Wocjan, P.: Efficient circuits for quantum walks. Quantum Inf. Comput. **10**(5–6), 420–434 (2010)
78. Childs, A.M.: Universal computation by quantum walk. Phys. Rev. Lett. **102**, 180501 (2009)
79. Childs, A.M.: On the relationship between continuous- and discrete-time quantum walk. Commun. Math. Phys. **294**(2), 581–603 (2010)
80. Childs, A.M., Eisenberg, J.M.: Quantum algorithms for subset finding. Quantum Inf. Comput. **5**(7), 593–604 (2005)
81. Childs, A.M., Farhi, E., Gutmann, S.: An example of the difference between quantum and classical random walks. Quantum Inf. Process. **1**(1), 35–43 (2002)
82. Childs, A.M., Goldstone, J.: Spatial search and the Dirac equation. Phys. Rev. A **70**, 042312 (2004)
83. Christandl, M., Datta, N., Ekert, A., Landahl, A.J.: Perfect state transfer in quantum spin networks. Phys. Rev. Lett. **92**, 187902 (2004)
84. Cohen-Tannoudji, C., Diu, B., Laloe, F.: Quantum Mechanics. Wiley-Interscience, New York (2006)
85. Connelly, E., Grammel, N., Kraut, M., Serazo, L., Tamon, C.: Universality in perfect state transfer. Linear Algebra Appl. **531**, 516–532 (2017)
86. Coutinho, G.: Quantum walks and the size of the graph. Discret. Math. (2018). arXiv:1802.08734
87. Coutinho, G., Godsil, C.: Perfect state transfer in products and covers of graphs. Linear Multilinear Algebra **64**(2), 235–246 (2016)
88. Coutinho, G., Godsil, C.: Perfect state transfer is poly-time. Quantum Inf. Comput. **17**(5–6), 495–502 (2017)
89. Coutinho, G., Portugal, R.: Discretization of continuous-time quantum walks via the staggered model with Hamiltonians. Nat. Comput. (2018). arXiv:1701.03423
90. Cover, T.M., Thomas, J.: Elements of Information Theory. Wiley, New York (1991)
91. Crespi, A., Osellame, R., Ramponi, R., Giovannetti, V., Fazio, R., Sansoni, L., De Nicola, F., Sciarrino, F., Mataloni, P.: Anderson localization of entangled photons in an integrated quantum walk. Nat. Photonics **7**(4), 322–328 (2013)
92. D'Ariano, G.M., Erba, M., Perinotti, P., Tosini, A.: Virtually abelian quantum walks. J. Phys. A: Math. Theor. **50**(3), 035301 (2017)
93. de Abreu, A.S.: Tesselações em grafos e suas aplicações em computação quântica. Master's thesis, UFRJ (2017)
94. de Abreu, A.S., Ferreira, M.M., Kowada, L.A.B., Marquezino, F.L.: QEDS: A classical simulator for quantum element distinctness. Revista de Informática Teórica e Aplicada **23**(2), 51–66 (2016)

95. Derevyanko, S.: Anderson localization of a one-dimensional quantum walker. Sci. Rep. **8**(1), 1795 (2018)
96. d'Espagnat, B.: Conceptual Foundations of Quantum Mechanics. Westview Press, Boulder (1999)
97. Desurvire, E.: Classical and Quantum Information Theory: An Introduction for the Telecom Scientist. Cambridge University Press, Cambridge (2009)
98. Dheeraj, M.N., Brun, T.A.: Continuous limit of discrete quantum walks. Phys. Rev. A **91**, 062304 (2015)
99. di Molfetta, G., Debbasch, F.: Discrete-time quantum walks: continuous limit and symmetries. J. Math. Phys. **53**(12), 123302 (2012)
100. Diao, Z., Zubairy, M.S., Chen, G.: A quantum circuit design for Grover's algorithm. Z. Naturforsch. A **57**(8), 701–708 (2002)
101. Díaz, N., Donangelo, R., Portugal, R., Romanelli, A.: Transient temperature and mixing times of quantum walks on cycles. Phys. Rev. A **94**, 012305 (2016)
102. Diestel, R.: Graph Theory. Graduate Texts in Mathematics, vol. 173. Springer, New York (2012)
103. Dodd, J.L., Ralph, T.C., Milburn, G.J.: Experimental requirements for Grover's algorithm in optical quantum computation. Phys. Rev. A **68**, 042328 (2003)
104. Domino, K., Glos, A., Ostaszewski, M.: Superdiffusive quantum stochastic walk definable on arbitrary directed graph. Quantum Info. Comput. **17**(11–12), 973–986 (2017)
105. Du, Y.-M., Lu, L.-H., Li, Y.-Q.: A rout to protect quantum gates constructed via quantum walks from noises. Sci. Rep. **8**(1), 7117 (2018)
106. Dukes, P.R.: Quantum state revivals in quantum walks on cycles. Results Phys. **4**, 189–197 (2014)
107. Dunjko, V., Briegel, H.J.: Quantum mixing of Markov chains for special distributions. New J. Phys. **17**(7), 073004 (2015)
108. Ellinas, D., Konstandakis, C.: Parametric quantum search algorithm as quantum walk: a quantum simulation. Rep. Math. Phys. **77**(1), 105–128 (2016)
109. Endo, T., Konno, N., Obuse, H., Segawa, E.: Sensitivity of quantum walks to a boundary of two-dimensional lattices: approaches based on the CGMV method and topological phases. J. Phys. A: Math. Theor. **50**(45), 455302 (2017)
110. Ermakov, V.L., Fung, B.M.: Experimental realization of a continuous version of the Grover algorithm. Phys. Rev. A **66**, 042310 (2002)
111. Falk, M.: Quantum search on the spatial grid (2013). arXiv:1303.4127
112. Falloon, P.E., Rodriguez, J., Wang, J.B.: QSWalk: a mathematica package for quantum stochastic walks on arbitrary graphs. Comput. Phys. Commun. **217**, 162–170 (2017)
113. Farhi, E., Gutmann, S.: Quantum computation and decision trees. Phys. Rev. A **58**, 915–928 (1998)
114. Feller, W.: An Introduction to Probability Theory and Its Applications. Wiley, New York (1968)
115. Foulger, I., Gnutzmann, S., Tanner, G.: Quantum walks and quantum search on graphene lattices. Phys. Rev. A **91**, 062323 (2015)
116. Fujiwara, S., Hasegawa, S.: General method for realizing the conditional phase-shift gate and a simulation of Grover's algorithm in an ion-trap system. Phys. Rev. A **71**, 012337 (2005)
117. Galiceanu, M., Strunz, W.T.: Continuous-time quantum walks on multilayer dendrimer networks. Phys. Rev. E **94**, 022307 (2016)
118. Mc Gettrick, M., Miszczak, J.A.: Quantum walks with memory on cycles. Phys. A **399**, 163–170 (2014)
119. Ghosh, J.: Simulating Anderson localization via a quantum walk on a one-dimensional lattice of superconducting qubits. Phys. Rev. A **89**, 022309 (2014)
120. Glos, A., Miszczak, J.A., Ostaszewski, M.: QSWalk.jl: Julia package for quantum stochastic walks analysis (2018). arXiv:1801.01294
121. Godsil, C., Royle, G.F.: Algebraic Graph Theory. Graduate Texts in Mathematics, vol. 207. Springer, New York (2001)

122. Gönülol, M., Aydner, E., Shikano, Y., Müstecaplolu, Ö.E.: Survival probability in a quantum walk on a one-dimensional lattice with partially absorbing traps. J. Comput. Theor. Nanosci. **10**(7), 1596–1600 (2013)
123. Gould, H.W.: Combinatorial Identities. Morgantown Printing and Binding Co., Morgantown (1972)
124. Goyal, S.K., Konrad, T., Diósi, L.: Unitary equivalence of quantum walks. Phys. Lett. A **379**(3), 100–104 (2015)
125. Graham, R.L., Knuth, D.E., Patashnik, O.: Concrete Mathematics: A Foundation for Computer Science. Addison-Wesley, Reading (1994)
126. Griffiths, D.: Introduction to Quantum Mechanics. Benjamin Cummings, Menlo Park (2005)
127. Grigoriev, D.: Karpinski, M., auf der Heide, F.M., Smolensky, R.: A lower bound for randomized algebraic decision trees. Comput. Complex. **6**(4), 357–375 (1996)
128. Grover, L.K.: A fast quantum mechanical algorithm for database search. In: Proceedings of the 28th Annual ACM Symposium on Theory of Computing, STOC '96, pp. 212–219. New York (1996)
129. Grover, L.K.: Quantum computers can search arbitrarily large databases by a single query. Phys. Rev. Lett. **79**(23), 4709–4712 (1997)
130. Grover, L.K.: Quantum mechanics helps in searching for a needle in a haystack. Phys. Rev. Lett. **79**(2), 325–328 (1997)
131. Grover, L.K.: Quantum computers can search rapidly by using almost any transformation. Phys. Rev. Lett. **80**(19), 4329–4332 (1998)
132. Grover, L.K.: Trade-offs in the quantum search algorithm. Phys. Rev. A **66**, 052314 (2002)
133. Gudder, S.: Quantum Probability. Academic Press, San Diego (1988)
134. Lo Gullo, N., Ambarish, C.V., Busch, T., Dell'Anna, L., Chandrashekar, C.M., Chandrashekar, C.M.: Dynamics and energy spectra of aperiodic discrete-time quantum walks. Phys. Rev. E **96**, 012111 (2017)
135. Hamada, M., Konno, N., Segawa, E.: Relation between coined quantum walks and quantum cellular automata. RIMS Kokyuroku **1422**, 1–11 (2005)
136. Hamilton, C.S., Barkhofen, S., Sansoni, L., Jex, I., Silberhorn, C.: Driven discrete time quantum walks. New J. Phys. **18**(7), 073008 (2016)
137. Hao, L., Li, J., Long, G.: Eavesdropping in a quantum secret sharing protocol based on Grover algorithm and its solution. Sci. China Phys. Mech. Astron. **53**(3), 491–495 (2010)
138. Harary, F.: Graph Theory. Addison-Wesley, Boston (1969)
139. Hartsfield, N., Ringel, G.: Pearls in Graph Theory: A Comprehensive Introduction. Dover Books on Mathematics. Dover Publications, New York (1994)
140. Hayashi, M., Ishizaka, S., Kawachi, A., Kimura, G., Ogawa, T.: Introduction to Quantum Information Science. Springer, Berlin (2014)
141. He, Z., Huang, Z., Li, L., Situ, H.: Coherence of one-dimensional quantum walk on cycles. Quantum Inf. Process. **16**(11), 271 (2017)
142. He, Z., Huang, Z., Li, L., Situ, H.: Coherence evolution in two-dimensional quantum walk on lattice. Int. J. Quantum Inf. **16**(02), 1850011 (2018)
143. He, Z., Huang, Z., Situ, H.: Dynamics of quantum coherence in two-dimensional quantum walk on finite lattices. Eur. Phys. J. Plus **132**(7), 299 (2017)
144. Hein, B., Tanner, G.: Quantum search algorithms on a regular lattice. Phys. Rev. A **82**(1), 012326 (2010)
145. Higuchi, Y., Konno, N., Sato, I., Segawa, E.: Spectral and asymptotic properties of Grover walks on crystal lattices. J. Funct. Anal. **267**(11), 4197–4235 (2014)
146. Hirvensalo, M.: Quantum Computing. Springer, Berlin (2010)
147. Ho, C.-L., Ide, Y., Konno, N., Segawa, E., Takumi, K.: A spectral analysis of discrete-time quantum walks related to the birth and death chains. J. Stat. Phys. **171**(2), 207–219 (2018)
148. Hoffman, K.M., Kunze, R.: Linear Algebra. Prentice Hall, New York (1971)
149. Holweck, F., Jaffali, H., Nounouh, I.: Grover's algorithm and the secant varieties. Quantum Inf. Process. **15**(11), 4391–4413 (2016)
150. Horn, R., Johnson, C.R.: Matrix Analysis. Cambridge University Press, Cambridge (1985)

151. Høyer, P.: Arbitrary phases in quantum amplitude amplification. Phys. Rev. A **62**, 052304 (2000)
152. Høyer, P., Komeili, M.: Efficient quantum walk on the grid with multiple marked elements. In: Vollmer, H., Vallée, B. (eds.) Proceedings of the 34th Symposium on Theoretical Aspects of Computer Science STACS, vol. 66, pp. 42:1–14. Dagstuhl, Germany (2017)
153. Hsu, L.-Y.: Quantum secret-sharing protocol based on Grover's algorithm. Phys. Rev. A **68**, 022306 (2003)
154. Hughes, B.D.: Random Walks and Random Environments: Voloum 1 Random Walks. Oxford University Press, Oxford (1995)
155. Hughes, B.D.: Random Walks and Random Environments: Volume 2 Random Environments. Oxford University Press, Oxford (1996)
156. Hush, M.R., Bentley, C.D.B., Ahlefeldt, R.L., James, M.R., Sellars, M.J., Ugrinovskii, V.: Quantum state transfer through time reversal of an optical channel. Phys. Rev. A **94**, 062302 (2016)
157. Ide, Y., Konno, N., Segawa, E.: Time averaged distribution of a discrete-time quantum walk on the path. Quantum Inf. Process. **11**(5), 1207–1218 (2012)
158. Innocenti, L., Majury, H., Giordani, T., Spagnolo, N., Sciarrino, F., Paternostro, M., Ferraro, A.: Quantum state engineering using one-dimensional discrete-time quantum walks. Phys. Rev. A **96**, 062326 (2017)
159. Itakura, Y.K.: Quantum algorithm for commutativity testing of a matrix set. Master's thesis, University of Waterloo, Waterloo (2005)
160. Iwai, T., Hayashi, N., Mizobe, K.: The geometry of entanglement and Grover's algorithm. J. Phys. A: Math. Theor. **41**(10), 105202 (2008)
161. Izaac, J.A., Wang, J.B.: PyCTQW: a continuous-time quantum walk simulator on distributed memory computers. Comput. Phys. Commun. **186**, 81–92 (2015)
162. Izaac, J.A., Wang, J.B.: Systematic dimensionality reduction for continuous-time quantum walks of interacting fermions. Phys. Rev. E **96**, 032136 (2017)
163. Izaac, J.A., Zhan, X., Bian, Z., Wang, K., Li, J., Wang, J.B., Xue, P.: Centrality measure based on continuous-time quantum walks and experimental realization. Phys. Rev. A **95**, 032318 (2017)
164. Jeffery, S., Magniez, F., de Wolf, R.: Optimal parallel quantum query algorithms. Algorithmica **79**(2), 509–529 (2017)
165. Jin, W., Chen, X.: A desired state can not be found with certainty for Grover's algorithm in a possible three-dimensional complex subspace. Quantum Inf. Process. **10**(3), 419–429 (2011)
166. Johnston, N., Kirkland, S., Plosker, S., Storey, R., Zhang, X.: Perfect quantum state transfer using Hadamard diagonalizable graphs. Linear Algebra Appl. **531**, 375–398 (2017)
167. Jones, J.A., Mosca, M., Hansen, R.H.: Implementation of a quantum search algorithm on a quantum computer. Nature **393**(6683), 344–346 (1998)
168. Kaplan, M.: Quantum attacks against iterated block ciphers. Mat. Vopr. Kriptogr. **7**, 71–90 (2016)
169. Kargin, V.: Bounds for mixing time of quantum walks on finite graphs. J. Phys. A: Math. Theor. **43**(33), 335302 (2010)
170. Kaye, P., Laflamme, R., Mosca, M.: An Introduction to Quantum Computing. Oxford University Press, New York (2007)
171. Kedon, V.M., Tamon, C.: Perfect state transfer in quantum walks on graphs. J. Comput. Theor. Nanosci. **8**(3), 422–433 (2011)
172. Kempe, J.: Quantum random walks - an introductory overview. Contemp. Phys. **44**(4), 302–327 (2003)
173. Kempe, J.: Discrete quantum walks hit exponentially faster. Probab. Theory Relat. Fields **133**(2), 215–235 (2005). arXiv:quant-ph/0205083
174. Kempton, M., Lippner, G., Yau, S.-T.: Perfect state transfer on graphs with a potential. Quantum Inf. Comput. **17**(3–4), 303–327 (2017)
175. Kendon, V.: Decoherence in quantum walks - a review. Math. Struct. Comput. Sci. **17**(6), 1169–1220 (2007)

176. Kendon, V., Sanders, B.C.: Complementarity and quantum walks. Phys. Rev. A **71**, 022307 (2005)
177. Kirkland, S., Severini, S.: Spin-system dynamics and fault detection in threshold networks. Phys. Rev. A **83**, 012310 (2011)
178. Kitaev, A.Y., Shen, A.H., Vyalyi, M.N.: Classical and Quantum Computation. American Mathematical Society, Boston (2002)
179. Koch, D., Hillery, M.: Finding paths in tree graphs with a quantum walk. Phys. Rev. A **97**, 012308 (2018)
180. Kollár, B., Novotný, J., Kiss, T., Jex, I.: Percolation induced effects in two-dimensional coined quantum walks: analytic asymptotic solutions. New J. Phys. **16**(2), 023002 (2014)
181. Komatsu, T., Konno, N.: Stationary amplitudes of quantum walks on the higher-dimensional integer lattice. Quantum Inf. Process. **16**(12), 291 (2017)
182. Konno, N.: Quantum random walks in one dimension. Quantum Inf. Process. **1**(5), 345–354 (2002)
183. Konno, N.: Quantum walks. In: Franz, U., Schürmann, M. (eds.) Quantum Potential Theory. Lecture Notes in Mathematics, vol. 1954, pp. 309–452. Springer, Berlin (2008)
184. Konno, N., Ide, Y., Sato, I.: The spectral analysis of the unitary matrix of a 2-tessellable staggered quantum walk on a graph. Linear Algebra Appl. **545**, 207–225 (2018)
185. Konno, N., Mitsuhashi, H., Sato, I.: The discrete-time quaternionic quantum walk on a graph. Quantum Inf. Process. **15**(2), 651–673 (2016)
186. Konno, N., Obata, N., Segawa, E.: Localization of the Grover walks on spidernets and free Meixner laws. Commun. Math. Phys. **322**(3), 667–695 (2013)
187. Konno, N., Portugal, R., Sato, I., Segawa, E.: Partition-based discrete-time quantum walks. Quantum Inf. Process. **17**(4), 100 (2018)
188. Konno, N., Sato, I.: On the relation between quantum walks and zeta functions. Quantum Inf. Process. **11**(2), 341–349 (2012)
189. Konno, N., Sato, I., Segawa, E.: The spectra of the unitary matrix of a 2-tessellable staggered quantum walk on a graph. Yokohama Math. J. **62**, 52–87 (2017)
190. Košík, J.: Two models of quantum random walk. Central Eur. J. Phys. **4**, 556–573 (2003)
191. Košík, J., Bužek, V., Hillery, M.: Quantum walks with random phase shifts. Phys. Rev. A **74**, 022310 (2006)
192. Krawec, W.O.: History dependent quantum walk on the cycle with an unbalanced coin. Phys. A **428**, 319–331 (2015)
193. Krovi, H., Brun, T.A.: Quantum walks with infinite hitting times. Phys. Rev. A **74**, 042334 (2006)
194. Krovi, H., Brun, T.A.: Quantum walks on quotient graphs. Phys. Rev. A **75**, 062332 (2007)
195. Krovi, H., Magniez, F., Ozols, M., Roland, J.: Finding is as easy as detecting for quantum walks. Automata, Languages and Programming. Lecture Notes in Computer Science, vol. 6198, pp. 540–551. Springer, Berlin (2010)
196. Kutin, S.: Quantum lower bound for the collision problem with small range. Theory Comput. **1**, 29–36 (2005)
197. Laarhoven, T., Mosca, M., van de Pol, J.: Finding shortest lattice vectors faster using quantum search. Des. Codes Cryptogr. **77**(2), 375–400 (2015)
198. Lam, H.T., Szeto, K.Y.: Ramsauer effect in a one-dimensional quantum walk with multiple defects. Phys. Rev. A **92**, 012323 (2015)
199. Lang, S.: Linear Algebra. Undergraduate Texts in Mathematics and Technology. Springer, New York (1987)
200. Lang, S.: Introduction to Linear Algebra. Undergraduate Texts in Mathematics. Springer, New York (1997)
201. Lara, P.C.S., Leão, A., Portugal, R.: Simulation of quantum walks using HPC. J. Comput. Int. Sci. **6**, 21 (2015)
202. Lehman, L.: Environment-induced mixing processes in quantum walks. Int. J. Quantum Inf. **12**(04), 1450021 (2014)

203. Leuenberger, M.N., Loss, D.: Grover algorithm for large nuclear spins in semiconductors. Phys. Rev. B **68**, 165317 (2003)
204. Li, D., Mc Gettrick, M., Gao, F., Xu, J., Wen, Q.-Y., Wen, Q.-Y.: Generic quantum walks with memory on regular graphs. Phys. Rev. A **93**, 042323 (2016)
205. Li, D., Mc Gettrick, M., Zhang, W.-W., Zhang, K.-J., Zhang, K.-J.: Quantum walks on two kinds of two-dimensional models. Int. J. Theor. Phys. **54**(8), 2771–2783 (2015)
206. Li, P., Li, S.: Phase matching in Grover's algorithm. Phys. Lett. A **366**(1), 42–46 (2007)
207. Li, P., Li, S.: Two improvements in Grover's algorithm. Chin. J. Electron. **17**(1), 100–104 (2008)
208. Lin, J.-Y., Zhu, X., Wu, S.: Limitations of discrete-time quantum walk on a one-dimensional infinite chain. Phys. Lett. A **382**(13), 899–903 (2018)
209. Lipton, R.J., Regan, K.W.: Quantum Algorithms via Linear Algebra: A Primer. MIT Press, Boston (2014)
210. Liu, C., Balu, R.: Steady states of continuous-time open quantum walks. Quantum Inf. Process. **16**(7), 173 (2017)
211. Loke, T., Tang, J.W., Rodriguez, J., Small, M., Wang, J.B.: Comparing classical and quantum pageranks. Quantum Inf. Process. **16**(1), 25 (2016)
212. Loke, T., Wang, J.B.: Efficient quantum circuits for continuous-time quantum walks on composite graphs. J. Phys. A: Math. Theor. **50**(5), 055303 (2017)
213. Loke, T., Wang, J.B.: Efficient quantum circuits for Szegedy quantum walks. Ann. Phys. **382**, 64–84 (2017)
214. Long, G.L.: Grover algorithm with zero theoretical failure rate. Phys. Rev. A **64**, 022307 (2001)
215. Lovász, L.: Random walks on graphs: a survey. Comb. Paul Erdös Eighty **2**, 1–46 (1993)
216. Lovett, N.B., Cooper, S., Everitt, M., Trevers, M., Kendon, V.: Universal quantum computation using the discrete-time quantum walk. Phys. Rev. A **81**, 042330 (2010)
217. Lovett, N.B., Everitt, M., Heath, R.M., Kendon, V.: The quantum walk search algorithm: factors affecting efficiency. Math. Struct. Comput. Sci. **67**, 1–141 (2018)
218. Lu, X., Yuan, J., Zhang, W.: Workflow of the Grover algorithm simulation incorporating CUDA and GPGPU. Comput. Phys. Commun. **184**(9), 2035–2041 (2013)
219. Ma, B.-W., Bao, W.-S., Li, T., Li, F.-G., Zhang, S., Fu, X.-Q.: A four-phase improvement of Grover's algorithm. Chin. Phys. Lett. **34**(7), 070305 (2017)
220. Machida, T., Chandrashekar, C.M.: Localization and limit laws of a three-state alternate quantum walk on a two-dimensional lattice. Phys. Rev. A **92**, 062307 (2015)
221. Machida, T., Chandrashekar, C.M., Konno, N., Busch, T.: Limit distributions for different forms of four-state quantum walks on a two-dimensional lattice. Quantum Inf. Comput. **15**(13–14), 1248–1258 (2015)
222. Mackay, T.D., Bartlett, S.D., Stephenson, L.T., Sanders, B.C.: Quantum walks in higher dimensions. J. Phys. A: Math. Gen. **35**(12), 2745 (2002)
223. Magniez, F., Nayak, A.: Quantum complexity of testing group commutativity. Algorithmica **48**(3), 221–232 (2007)
224. Magniez, F., Nayak, A., Richter, P., Santha, M.: On the hitting times of quantum versus random walks. In: Proceedings of the 20th ACM-SIAM Symposium on Discrete Algorithms (2009)
225. Magniez, F., Nayak, A., Roland, J., Santha, M.: Search via quantum walk. In: Proceedings of the 39th ACM Symposium on Theory of Computing, pp. 575–584 (2007)
226. Magniez, F., Santha, M., Szegedy, M.: Quantum algorithms for the triangle problem. SIAM J. Comput. **37**(2), 413–424 (2007)
227. Makmal, A., Tiersch, M., Ganahl, C., Briegel, H.J.: Quantum walks on embedded hypercubes: nonsymmetric and nonlocal cases. Phys. Rev. A **93**, 022322 (2016)
228. Makmal, A., Zhu, M., Manzano, D., Tiersch, M., Briegel, H.J.: Quantum walks on embedded hypercubes. Phys. Rev. A **90**, 022314 (2014)
229. Manouchehri, K., Wang, J.: Physical Implementation of Quantum Walks. Springer, New York (2014)

230. Marinescu, D.C., Marinescu, G.M.: Approaching Quantum Computing. Pearson/Prentice Hall, Michigan (2005)
231. Marquezino, F.L., Portugal, R.: The qwalk simulator of quantum walks. Comput. Phys. Commun. **179**(5), 359–369 (2008)
232. Marquezino, F.L., Portugal, R., Abal, G.: Mixing times in quantum walks on two-dimensional grids. Phys. Rev. A **82**(4), 042341 (2010)
233. Marquezino, F.L., Portugal, R., Abal, G., Donangelo, R.: Mixing times in quantum walks on the hypercube. Phys. Rev. A **77**, 042312 (2008)
234. Mermin, N.D.: Quantum Computer Science: An Introduction. Cambridge University Press, New York (2007)
235. Meyer, C.D.: Matrix Analysis and Applied Linear Algebra. SIAM, Philadelphia (2001)
236. Meyer, D.A.: From quantum cellular automata to quantum lattice gases. J. Stat. Phys. **85**(5), 551–574 (1996)
237. Meyer, D.A.: Sophisticated quantum search without entanglement. Phys. Rev. Lett. **85**, 2014–2017 (2000)
238. Meyer, D.A., Wong, T.G.: Connectivity is a poor indicator of fast quantum search. Phys. Rev. Lett. **114**, 110503 (2015)
239. Mlodinow, L., Brun, T.A.: Discrete spacetime, quantum walks, and relativistic wave equations. Phys. Rev. A **97**, 042131 (2018)
240. Moore, C., Mertens, S.: The Nature of Computation. Oxford University Press, New York (2011)
241. Moore, C., Russell, A.: Quantum walks on the hypercube. In: Proceedings of the 6th International Workshop on Randomization and Approximation Techniques RANDOM, pp. 164–178. Springer, Berlin (2002)
242. Moqadam, J.K., Portugal, R., de Oliveira, M.C.: Quantum walks on a circle with optomechanical systems. Quantum Inf. Process. **14**(10), 3595–3611 (2015)
243. Khatibi Moqadam, J., de Oliveira, M.C., Portugal, R., Portugal, R.: Staggered quantum walks with superconducting microwave resonators. Phys. Rev. B **95**, 144506 (2017)
244. Mosca, M.: Counting by quantum eigenvalue estimation. Theor. Comput. Sci. **264**(1), 139–153 (2001)
245. Motwani, R., Raghavan, P.: Randomized algorithms. ACM Comput. Surv. **28**(1), 33–37 (1996)
246. Mülken, O., Blumen, A.: Continuous-time quantum walks: models for coherent transport on complex networks. Phys. Rep. **502**(2), 37–87 (2011)
247. Nayak, A., Vishwanath, A.: Quantum walk on a line. DIMACS Technical Report 2000-43 (2000). arXiv:quant-ph/0010117
248. Nielsen, M.A., Chuang, I.L.: Quantum Computation and Quantum Information. Cambridge University Press, New York (2000)
249. Novo, L., Chakraborty, S., Mohseni, M., Neven, H., Omar, Y.: Systematic dimensionality reduction for quantum walks: optimal spatial search and transport on non-regular graphs. Sci. Rep. **5**, 13304 (2015)
250. Ohno, H.: Unitary equivalent classes of one-dimensional quantum walks. Quantum Inf. Process. **15**(9), 3599–3617 (2016)
251. Oliveira, A.C., Portugal, R., Donangelo, R.: Decoherence in two-dimensional quantum walks. Phys. Rev. A **74**, 012312 (2006)
252. Omnès, R.: Understanding Quantum Mechanics. Princeton University Press, Princeton (1999)
253. Pal, H., Bhattacharjya, B.: Perfect state transfer on gcd-graphs. Linear Multilinear Algebra **65**(11), 2245–2256 (2017)
254. Paparo, G.D., Müller, M., Comellas, F., Martin-Delgado, M.A.: Quantum Google in a complex network. Sci. Rep. **3**, 2773 (2013)
255. Parthasarathy, K.R.: The passage from random walk to diffusion in quantum probability. J. Appl. Probab. **25**, 151–166 (1988)
256. Patel, A., Raghunathan, K.S., Rungta, P.: Quantum random walks do not need a coin toss. Phys. Rev. A **71**, 032347 (2005)
257. Peres, A.: Quantum Theory: Concepts and Methods. Springer, Berlin (1995)

258. Pérez, A., Romanelli, A.: Spatially dependent decoherence and anomalous diffusion of quantum walks. J. Comput. Theor. Nanosci. **10**(7), 1591–1595 (2013)
259. Peterson, D.: Gridline graphs: a review in two dimensions and an extension to higher dimensions. Discret. Appl. Math. **126**(2), 223–239 (2003)
260. Philipp, P., Portugal, R.: Exact simulation of coined quantum walks with the continuous-time model. Quantum Inf. Process. **16**(1), 14 (2017)
261. Philipp, P., Tarrataca, L., Boettcher, S.: Continuous-time quantum search on balanced trees. Phys. Rev. A **93**, 032305 (2016)
262. Piccinini, E., Benedetti, C., Siloi, I., Paris, M.G.A., Bordone, P.: GPU-accelerated algorithms for many-particle continuous-time quantum walks. Comput. Phys. Commun. **215**, 235–245 (2017)
263. Portugal, R.: Establishing the equivalence between Szegedy's and coined quantum walks using the staggered model. Quantum Inf. Process. **15**(4), 1387–1409 (2016)
264. Portugal, R.: Staggered quantum walks on graphs. Phys. Rev. A **93**, 062335 (2016)
265. Portugal, R.: Element distinctness revisited. Quantum Inf. Process. **17**(7), 163 (2018)
266. Portugal, R., Boettcher, S., Falkner, S.: One-dimensional coinless quantum walks. Phys. Rev. A **91**, 052319 (2015)
267. Portugal, R., de Oliveira, M.C., Moqadam, J.K.: Staggered quantum walks with Hamiltonians. Phys. Rev. A **95**, 012328 (2017)
268. Portugal, R., Fernandes, T.D.: Quantum search on the two-dimensional lattice using the staggered model with Hamiltonians. Phys. Rev. A **95**, 042341 (2017)
269. Portugal, R., Santos, R.A.M., Fernandes, T.D., Gonçalves, D.N.: The staggered quantum walk model. Quantum Inf. Process. **15**(1), 85–101 (2016)
270. Portugal, R., Segawa, E.: Connecting coined quantum walks with Szegedy's model. Interdiscip. Inf. Sci. **23**(1), 119–125 (2017)
271. Preiss, P.M., Ma, R., Tai, M.E., Lukin, A., Rispoli, M., Zupancic, P., Lahini, Y., Islam, R., Greiner, M.: Strongly correlated quantum walks in optical lattices. Science **347**(6227), 1229–1233 (2015)
272. Preskill, J.: Lecture Notes on Quantum Computation (1998). http://www.theory.caltech.edu/~preskill/ph229
273. Reitzner, D., Hillery, M., Koch, D.: Finding paths with quantum walks or quantum walking through a maze. Phys. Rev. A **96**, 032323 (2017)
274. Reitzner, D., Nagaj, D., Bužek, V.: Quantum walks. Acta Phys. Slov. **61**(6), 603–725 (2011)
275. Ren, P., Aleksić, T., Emms, D., Wilson, R.C., Hancock, E.R.: Quantum walks, Ihara zeta functions and cospectrality in regular graphs. Quantum Inf. Process. **10**(3), 405–417 (2011)
276. Rieffel, E., Polak, W.: Quantum Computing, a Gentle Introduction. MIT Press, Cambridge (2011)
277. Robens, C., Alt, W., Meschede, D., Emary, C., Alberti, A.: Ideal negative measurements in quantum walks disprove theories based on classical trajectories. Phys. Rev. X **5**, 011003 (2015)
278. Rodrigues, J., Paunković, N., Mateus, P.: A simulator for discrete quantum walks on lattices. Int. J. Mod. Phys. C **28**(04), 1750055 (2017)
279. Rohde, P.P., Fedrizzi, A., Ralph, T.C.: Entanglement dynamics and quasi-periodicity in discrete quantum walks. J. Modern Opt. **59**(8), 710–720 (2012)
280. Romanelli, A., Siri, R., Abal, G., Auyuanet, A., Donangelo, R.: Decoherence in the quantum walk on the line. Phys. A **347**, 137–152 (2005)
281. Ronke, R., Estarellas, M.P., D'Amico, I., Spiller, T.P., Miyadera, T.: Anderson localisation in spin chains for perfect state transfer. Eur. Phys. J. D **70**(9), 189 (2016)
282. Rosmanis, A.: Quantum adversary lower bound for element distinctness with small range. Chic. J. Theor. Comput. Sci. **4**, 2014 (2014)
283. Rossi, M.A.C., Benedetti, C., Borrelli, M., Maniscalco, S., Paris, M.G.A.: Continuous-time quantum walks on spatially correlated noisy lattices. Phys. Rev. A **96**, 040301 (2017)
284. Rudinger, K., Gamble, J.K., Bach, E., Friesen, M., Joynt, R., Coppersmith, S.N.: Comparing algorithms for graph isomorphism using discrete- and continuous-time quantum random walks. J. Comput. Theor. Nanosci. **10**(7), 1653–1661 (2013)

285. Sadgrove, M.: Quantum amplitude amplification by phase noise. Europhys. Lett. **86**(5), 50005 (2009)
286. Sadowski, P., Miszczak, J.A., Ostaszewski, M.: Lively quantum walks on cycles. J. Phys. A: Math. Theor. **49**(37), 375302 (2016)
287. Sakurai, J.J., Napolitano, J.: Modern Quantum Mechanics. Addison-Wesley, Reading (2011)
288. Salas, P.J.: Noise effect on Grover algorithm. Eur. Phys. J. D **46**(2), 365–373 (2008)
289. Sampathkumar, E.: On the line graph of subsivision graph. J. Karnatak Univ. Sci. **17**, 259–260 (1972)
290. Santha, M.: Quantum walk based search algorithms. In: Proceedings of the 5th International Conference, TAMC 2008, Xi'an, China, 2008, pp. 31–46. Springer, Berlin (2008)
291. Santos, R.A.M.: Szegedy's quantum walk with queries. Quantum Inf. Process. **15**(11), 4461–4475 (2016)
292. Santos, R.A.M., Portugal, R.: Quantum hitting time on the complete graph. Int. J. Quantum Inf. **8**(5), 881–894 (2010)
293. Santos, R.A.M., Portugal, R., Boettcher, S.: Moments of coinless quantum walks on lattices. Quantum Inf. Process. **14**(9), 3179–3191 (2015)
294. Schmitz, A.T., Schwalm, W.A.: Simulating continuous-time hamiltonian dynamics by way of a discrete-time quantum walk. Phys. Lett. A **380**(11), 1125–1134 (2016)
295. Segawa, E.: Localization of quantum walks induced by recurrence properties of random walks. J. Comput. Theor. Nanosci. **10**(7), 1583–1590 (2013)
296. Shakeel, A., Meyer, D.A., Love, P.J.: History dependent quantum random walks as quantum lattice gas automata. J. Math. Phys. **55**(12), 122204 (2014)
297. Shenvi, N., Kempe, J., Whaley, K.B.: A quantum random walk search algorithm. Phys. Rev. A **67**(5), 052307 (2003)
298. Shi, H.-L., Liu, S.-Y., Wang, X.-H., Yang, W.-L., Yang, Z.-Y., Fan, H.: Coherence depletion in the Grover quantum search algorithm. Phys. Rev. A **95**, 032307 (2017)
299. Souza, A.M.C., Andrade, R.F.S.: Discrete time quantum walk on the Apollonian network. J. Phys. A: Math. Theor. **46**(14), 145102 (2013)
300. Souza, D.S., Marquezino, F.L., Lima, A.A.B.: Quandoop: a classical simulator of quantum walks on computer clusters. J. Comput. Int. Sci. **8**, 109–172 (2017)
301. Stefaňák, M., Skoupý, S.: Perfect state transfer by means of discrete-time quantum walk on complete bipartite graphs. Quantum Inf. Process. **16**(3), 72 (2017)
302. Stefaňák, M., Kollár, B., Kiss, T., Jex, I.: Full revivals in 2D quantum walks. Phys. Scr. **2010**(T140), 014035 (2010)
303. Stolze, J., Suter, D.: Quantum Computing, Revised and Enlarged: A Short Course from Theory to Experiment. Wiley-VCH, New York (2008)
304. Stong, R.A.: On 1-factorizability of Cayley graphs. J. Comb. Theory Ser. B **39**(3), 298–307 (1985)
305. Strang, G.: Linear Algebra and Its Applications. Brooks Cole (1988)
306. Strauch, F.W.: Connecting the discrete- and continuous-time quantum walks. Phys. Rev. A **74**(3), 030301 (2006)
307. Szegedy, M.: Quantum speed-up of Markov chain based algorithms. In: Proceedings of the 45th Annual IEEE Symposium on Foundations of Computer Science, FOCS '04, pp. 32–41. Washington (2004)
308. Szegedy, M.: Spectra of quantized walks and a $\sqrt{\delta\epsilon}$ rule (2004). arXiv:quant-ph/0401053
309. Tani, S.: Claw finding algorithms using quantum walk. Theor. Comput. Sci. **410**(50), 5285–5297 (2009)
310. Tödtli, B., Laner, M., Semenov, J., Paoli, B., Blattner, M., Kunegis, J.: Continuous-time quantum walks on directed bipartite graphs. Phys. Rev. A **94**, 052338 (2016)
311. Toyama, F.M., Van Dijk, W., Nogami, Y.: Quantum search with certainty based on modified Grover algorithms: optimum choice of parameters. Quantum Inf. Process. **12**(5), 1897–1914 (2013)
312. Travaglione, B.C., Milburn, G.J.: Implementing the quantum random walk. Phys. Rev. A **65**(3), 032310 (2002)

313. Tregenna, B., Flanagan, W., Maile, R., Kendon, V.: Controlling discrete quantum walks: coins and initial states. New J. Phys. **5**(1), 83 (2003)
314. Trudeau, R.J.: Introduction to Graph Theory. Dover Books on Mathematics. Dover Publications, Mineola (2013)
315. Tulsi, A.: Faster quantum-walk algorithm for the two-dimensional spatial search. Phys. Rev. A **78**(1), 012310 (2008)
316. Tulsi, A.: General framework for quantum search algorithms. Phys. Rev. A **86**, 042331 (2012)
317. Tulsi, A.: Robust quantum spatial search. Quantum Inf. Process. **15**(7), 2675–2683 (2016)
318. Valencia-Pabon, M., Vera, J.-C.: On the diameter of Kneser graphs. Discret. Math. **305**(1), 383–385 (2005)
319. Venegas-Andraca, S.E.: Quantum Walks for Computer Scientists. Morgan and Claypool Publishers (2008)
320. Venegas-Andraca, S.E.: Quantum walks: a comprehensive review. Quantum Inf. Process. **11**(5), 1015–1106 (2012)
321. Vieira, R., Amorim, E.P.M., Rigolin, G.: Entangling power of disordered quantum walks. Phys. Rev. A **89**, 042307 (2014)
322. Wang, C., Hao, L., Song, S.Y., Long, G.L.: Quantum direct communication based on quantum search algorithm. Int. J. Quantum Inf. **08**(03), 443–450 (2010)
323. Wang, G.: Efficient quantum algorithms for analyzing large sparse electrical networks. Quantum Inf. Comput. **17**(11&12), 987–1026 (2017)
324. Wang, Y., Shang, Y., Xue, P.: Generalized teleportation by quantum walks. Quantum Inf. Process. **16**(9), 221 (2017)
325. Waseem, M., Ahmed, R., Irfan, M., Qamar, S.: Three-qubit Grover's algorithm using superconducting quantum interference devices in cavity-QED. Quantum Inf. Process. **12**(12), 3649–3664 (2013)
326. West, D.B.: Introduction to Graph Theory. Prentice Hall, Englewood Cliffs (2000)
327. Whitfield, J.D., Rodríguez-Rosario, C.A., Aspuru-Guzik, A.: Quantum stochastic walks: a generalization of classical random walks and quantum walks. Phys. Rev. A **81**, 022323 (2010)
328. Williams, C.P.: Explorations in Quantum Computing. Springer, Berlin (2008)
329. Wong, T.G.: Quantum walk search with time-reversal symmetry breaking. J. Phys. A: Math. Theor. **48**(40), 405303 (2015)
330. Wong, T.G.: Coined quantum walks on weighted graphs. J. Phys. A: Math. Theor. **50**(47), 475301 (2017)
331. Wong, T.G.: Faster search by lackadaisical quantum walk. Quantum Inf. Process. **17**(3), 68 (2018)
332. Wong, T.G., Ambainis, A.: Quantum search with multiple walk steps per oracle query. Phys. Rev. A **92**, 022338 (2015)
333. Wong, T.G., Santos, R.A.M.: Exceptional quantum walk search on the cycle. Quantum Inf. Process. **16**(6), 154 (2017)
334. Wong, T.G., Tarrataca, L., Nahimov, N.: Laplacian versus adjacency matrix in quantum walk search. Quantum Inf. Process. **15**(10), 4029–4048 (2016)
335. Xi-Ling, X., Zhi-Hao, L., Han-Wu, C.: Search algorithm on strongly regular graphs based on scattering quantum walk. Chin. Phys. B **26**(1), 010301 (2017)
336. Xu, X.-P., Ide, Y.: Exact solutions and symmetry analysis for the limiting probability distribution of quantum walks. Ann. Phys. **373**, 682–693 (2016)
337. Xu, X.-P., Ide, Y., Konno, N.: Symmetry and localization of quantum walks induced by an extra link in cycles. Phys. Rev. A **85**, 042327 (2012)
338. Yalçinkaya, I., Gedik, Z.: Two-dimensional quantum walk under artificial magnetic field. Phys. Rev. A **92**, 042324 (2015)
339. Yalouz, S., Pouthier, V.: Continuous-time quantum walk on an extended star graph: trapping and superradiance transition. Phys. Rev. E **97**, 022304 (2018)
340. Yamaguchi, F., Milman, P., Brune, M., Raimond, J.M., Haroche, S.: Quantum search with two-atom collisions in cavity QED. Phys. Rev. A **66**, 010302 (2002)

341. Yang-Yi, H., Ping-Xing, C.: Localization of quantum walks on finite graphs. Chin. Phys. B **25**(12), 120303 (2016)
342. Yao, A.C.C.: Near-optimal time-space tradeoff for element distinctness. In: Proceedings of the 29th Annual Symposium on Foundations of Computer Science, pp. 91–97 (1988)
343. Yoon, C.S., Kang, M.S., Lim, J.I., Yang, H.J.: Quantum signature scheme based on a quantum search algorithm. Phys. Scr. **90**(1), 015103 (2015)
344. Zalka, C.: Grover's quantum searching algorithm is optimal. Phys. Rev. A **60**, 2746–2751 (1999)
345. Zeng, M., Yong, E.H.: Discrete-time quantum walk with phase disorder: localization and entanglement entropy. Sci. Rep. **7**(1), 12024 (2017)
346. Zhan, X., Qin, H., Bian, Z.-H., Li, J., Xue, P.: Perfect state transfer and efficient quantum routing: a discrete-time quantum-walk approach. Phys. Rev. A **90**, 012331 (2014)
347. Zhang, F., Chen, Y.-C., Chen, Z.: Clique-inserted-graphs and spectral dynamics of clique-inserting. J. Math. Anal. Appl. **349**(1), 211–225 (2009)
348. Zhang, R., Qin, H., Tang, B., Xue, P.: Disorder and decoherence in coined quantum walks. Chin. Phys. B **22**(11), 110312 (2013)
349. Zhang, Y.-C., Bao, W.-S., Wang, X., Fu, X.-Q.: Effects of systematic phase errors on optimized quantum random-walk search algorithm. Chin. Phys. B **24**(6), 060304 (2015)
350. Zhang, Y.-C., Bao, W.-S., Wang, X., Fu, X.-Q.: Optimized quantum random-walk search algorithm for multi-solution search. Chin. Phys. B **24**(11), 110309 (2015)
351. Zhirov, O.V., Shepelyansky, D.L.: Dissipative decoherence in the Grover algorithm. Eur. Phys. J. D **38**(2), 405–408 (2006)
352. Zhou, J., Bu, C.: State transfer and star complements in graphs. Discret. Appl. Math. **176**, 130–134 (2014)

Index

R. Portugal, *Quantum Walks and Search Algorithms*, Quantum Science and Technology, https://doi.org/10.1007/978-3-319-97813-0