Decompositions of general quantum gates

Mikko Möttönen and Juha J. Vartiainen

Materials Physics Laboratory, POB 2200 (Technical Physics)
FIN-02015 HUT, Helsinki University of Technology, Finland

Abstract

Quantum algorithms may be described by sequences of unitary transformations called quantum gates and measurements applied to the quantum register of n quantum bits, qubits. A collection of quantum gates is called universal if it can be used to construct any n-qubit gate. In 1995, the universality of the set of one-qubit gates and controlled NOT gate was shown by Barenco *et al.* using QR decomposition of unitary matrices. Almost ten years later the decomposition was improved to include essentially fewer elementary gates. In addition, the cosine-sine matrix decomposition was applied to efficiently implement decompositions of general quantum gates. In this chapter, we review the different types of general gate decompositions and slightly improve the best known gate count for the controlled NOT gates to $\frac{23}{48}4^n$ in the leading order. In physical realizations, the interaction strength between the qubits can decrease strongly as a function of their distance. Therefore, we also discuss decompositions with the restriction to nearest-neighbor interactions in a linear chain of qubits.

1 Introduction

The emerging of quantum mechanics [1] in the beginning of the 20^{th} century revolutionized the field of physics bringing not only understanding to fundamental concepts such as atomic and particle physics, but also numerous applications for everyday life. One of the most important applications are the semiconductors, namely the transistor which is the basis of today's digital computers. As quantum mechanics shook up physics, quantum computing [2] has done the same for computer science. Some quantum algorithms, Shor's algorithm [3] being the most famous, offer exponential speedup compared with the best known classical counterparts due to the phenomenon called quantum parallelism. Shor's algorithm may be used to break the commonly used RSA encryption for key distribution but, on the other hand, quantum physics also provides a secure information channel using quantum

cryptography [4]. Due to its powerful applications, the experimental realization of the quantum computer is regarded as highly important issue in physics. Similarly, theoretical research which lightens the burden of the experimental needs is also of great interest.

In quantum computing, the algorithms are commonly described by the quantum circuit model [5]. It involves quantum gates, projective measurements and a register of n quantum bits, called qubits. In a classical computer, a bit may have only two distinct values usually denoted by 0 and 1. In contrast, a qubit may be in a superposition of these two basis vectors, *i.e.*, the state of the qubit is described by a vector in the complex space $\mathbb{C}^2$. In this space, the quantum gates correspond to matrices, which are unitary due to the unitary temporal evolution of any closed quantum system.

Since many algorithms involve gates acting on n qubits, it is an important issue how these gates may be decomposed into an array of simpler gates accessible to the experiments. In general, we may assume that we have a collection of simple quantum gates, called the gate library, into which the n-qubit gates are to be decomposed. The gates in the gate library are called elementary gates. The library is called universal if any n-qubit gate has a presentation only involving gates from that library. We choose our library to consist of all one-qubit gates and the controlled NOT gate (CNOT) which are defined in Sec. 2.2. This particular library has been proved to be universal [6] but, actually, almost any other two-qubit gate could be chosen to replace the CNOT for the universality to hold [7]. However, it is feasible to work with the CNOT since it has a rather simple logical structure.

The proof of the universality of our gate library [6] was, in fact, constructive but the number of CNOTs involved was as high as $O(n^3 4^n)$. It is convenient to calculate the number of CNOTs and one-qubit gates separately, since CNOTs introduce interactions between the qubits and those interactions are usually much weaker than the interactions between a single qubit and the control fields. Hence, the experimental realization of the CNOT is typically a much slower process than that of a one-qubit gate. Already in 1995, it was shown that the circuit complexity could be reduced down to $O(n4^n)$ [8], but until the year 2004 there was no remarkable progress on the decomposition of arbitrary quantum gates. Reference [9] reviews briefly the traditional decomposition of Ref. [6].

The highest known lower bound for the number of CNOTs needed to decompose a general unstructured quantum gate acting on n qubits is $\lceil (4^n - 3n - 1)/4 \rceil$ [10] and, hence, there was an extra factor of n in the best known complexity. Finally in 2004, being an unsolved mystery for about ten years the original construction was improved to yield the complexity $O(4^n)$ [11]. However, this decomposition was still far from the lower bound. The original gate decomposition made use of the QR matrix decomposition [12]. In contrast, Ref. [13] introduces the cosine-sine matrix decomposition[1] (CSD) [12] in this context which turned out to yield a leading order

[1] In context of quantum computing, the CSD was discussed first in Ref. [14].

complexity 4^n for the one-qubit gates and 4^n for the CNOTs. The CSD was also combined with a so-called quantum multiplexor (QM), a special method to simplify the gates, to obtain a decomposition involving $4^n/2$ CNOTs and the same number of one-qubit gates in the leading order [15] (see also Ref. [16]). In this chapter, we present an improvement to the decomposition introduced in Ref. [16] to obtain the lowest CNOT count known to date.

This chapter is organized as follows. In Sec. 2, we define our notation and introduce some of the important mathematical tools. Section 3 is devoted to the presentation of so-called uniformly controlled gates (UCGs) and their efficient decomposition into elementary gates. The UCGs are the natural building blocks of decompositions employing the CSD. The original QR decomposition and its improved versions are discussed in Sec. 4 in contrast to Sec. 5 in which the CSD is studied. Finally, the local state preparation, *i.e.*, the question how to transform any given quantum state into another arbitrary state, is implemented Sec. 6 following Refs. [15–17]. The state preparation may be useful if one wishes to use, for example, exotic inputs to algorithms. In Sec. 7, we conclude and summarize our discussions.

2 Preliminaries

2.1 Quantum state and unitary temporal evolution

We consider here a quantum register consisting of n qubits and, hence, all possible quantum states of the system are in the Hilbert space $\mathcal{H} := \bigotimes_{i=1}^{n} \mathbb{C}^2 = \mathbb{C}^{2^n}$, where the symbol $\otimes$ denotes the Kronecker product. The basis vectors for each of the qubits are chosen as

$$|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \quad \text{and} \quad |1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}. \tag{1}$$

For the whole configuration space $\mathcal{H}$, it is convenient to choose the basis vectors to be $\{|e_k\rangle\}$, $k = 1, \ldots, N := 2^n$. Here $|e_k\rangle = \bigotimes_i |x_i^k\rangle$, where $x_i^k \in \{0, 1\}$ and the index $i = 1, ..., n$ refers to the qubit. In this basis the state vector of the system is of the form

$$|\Psi\rangle = \sum_{i=1}^{N} a_i |e_i\rangle \quad \text{and} \quad \sum_{i=1}^{N} |a_i|^2 = 1, \tag{2}$$

where the latter equality fixes the normalization of the vector. Hence, the probability for the system to be in a state $|e_i\rangle$ after a projective measurement is $|a_i|^2$. It is also noted that the global phase of the state vector is unobservable and, hence, may be taken to unity[2].

[2]Clearly the global phase does not affect the probabilities. Furthermore, addition of a global phase commutes with any unitary matrix, *i.e.*, it has no effect on the temporal evolution of the system.

Conventionally in quantum computing, the order of the basis vectors has been chosen such that the values x_i^k essentially form the binary representation of the number $k-1$, *i.e.*, $k = 1 + \sum_{i=0}^{n} 2^i x_i^k$. We note that the order of the basis vectors in the computational basis can be freely chosen. We will make use of this degree of freedom in Sec. 4 in the context of the QR decomposition.

The fundamental differences of the quantum computer compared with the classical one arise from the utilization of the high-dimensional Hilbert space $\mathcal{H}$. In comparison, the states accessible to a classical computer are limited to the basis vectors $|\Psi\rangle = |e_i\rangle$, *i.e.*, to the states in which all of the weight factors except one vanish. The quantum mechanical superposition principle allows several weight factors to be simultaneously non-zero, which renders the quantum mechanical state space greatly larger than the classical one.

The temporal evolution of any quantum system is governed by the well known Schrödinger equation

$$i\hbar \frac{\partial}{\partial t} |\Phi(t)\rangle = H|\Phi(t)\rangle, \tag{3}$$

where the Hamiltonian H of the pure quantum system is always Hermitian. This implies that the temporal evolution may be described by a unitary operator $\mathcal{U}(t,0)$ as $|\Phi(t)\rangle = \mathcal{U}(t,0)|\Phi(0)\rangle$. In our finite dimensional Hilbert space the unitary operator may be written as a unitary matrix $U \in SU(N)$. The reason why the determinant of U may be taken to unity is that the global phase of the state vector has no physical meaning. Since the n-qubit quantum gate may be represented by a unitary matrix, it is reasonable that the gate decompositions may correspond to some known matrix decompositions and vice versa.

2.2 Quantum circuits

A one-qubit gate $U \in SU(2)$ acting on the k^{th} qubit in a n-qubit register is represented by a unitary matrix

$$\tilde{U} = \underbrace{I \otimes \ldots \otimes I}_{k-1 \text{ times}} \otimes U \underbrace{\otimes I \ldots \otimes I}_{n-k \text{ times}}, \tag{4}$$

For simplicity, we omit below the qubits that are operated on only by an identity operator. Accordingly, the matrix representation of the gate U is

$$U = \begin{pmatrix} a & b \\ -\bar{b} & \bar{a} \end{pmatrix}, \tag{5}$$

where a and b are two complex numbers satisfying $|a|^2 + |b|^2 = 1$. We fix the basis for the two-state system such that the generator σ_z of the $SU(2)$ group is diagonal. Furthermore, we call the vectors corresponding to the eigenvalues 1 and -1 by $|0\rangle$ and $|1\rangle$, respectively. In this basis the matrix representations of generators $\{\sigma_i\}$ are

called the Pauli spin matrices

$$\sigma_x = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad \sigma_y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}, \quad \sigma_z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}. \tag{6}$$

In fact, any $U \in SU(2)$ may be written as a rotation

$$U = R_{\mathbf{a}}(\theta) = e^{i\mathbf{a}\cdot\boldsymbol{\sigma}\theta/2} = I\cos\frac{\theta}{2} + i\,(\mathbf{a}\cdot\boldsymbol{\sigma})\sin\frac{\theta}{2}, \tag{7}$$

where the symbol θ stands for the rotation angle around the unit vector $\mathbf{a}$ fixed by U and we have introduced the product $\mathbf{a}\cdot\boldsymbol{\sigma} = a_x\sigma_x + a_y\sigma_y + a_z\sigma_z$. Equation (7) yields that any rotation $R_{\mathbf{a}}(\theta)$ can be made diagonal as

$$R_{\mathbf{a}}(\theta) = V_{\mathbf{a}} R_z(\theta) V_{\mathbf{a}}^{\dagger}, \tag{8}$$

where the similarity transformation $V_{\mathbf{a}}$ diagonalizes the matrix $\mathbf{a}\cdot\boldsymbol{\sigma}$. We note that the matrix $V_{\mathbf{a}}$ does not depend on the rotation angle θ. In addition, all rotations about any single axis are additive

$$R_{\mathbf{a}}(\theta_1)R_{\mathbf{a}}(\theta_2) = R_{\mathbf{a}}(\theta_1+\theta_2), \tag{9}$$

and the rotation angle of all rotations with $a_x = 0$ is reversed by conjugation with σ_x as

$$a_x = 0 \implies \sigma_x R_{\mathbf{a}}(\theta)\sigma_x = R_{\mathbf{a}}(-\theta). \tag{10}$$

The rotations for which the rotation vector is parallel to any of the coordinate axes are called elementary rotations and denoted by $R_x(\theta)$, $R_y(\theta)$ and $R_z(\theta)$. Any element $U \in SU(2)$ may be written using only two different types of elementary rotations, *e.g.*, z and y rotations as

$$U = R_z(\alpha)R_y(\beta)R_z(\gamma), \tag{11}$$

where angles α, β, γ are called the Euler angles. The above results are used in the next sections to achieve and simplify the studied gate decompositions.

The circuit diagram for the one-qubit gate U is shown in Fig. 1(a). The only two-qubit gate in out library is the CNOT shown in Fig. 1(b). The action of the CNOT is logical NOT in the subspace $\{|10\rangle, |11\rangle\}$ and it leaves the subspace where the value of the control qubit (the upper qubit) is zero untouched. The matrix presentation for the CNOT in basis $\{|00\rangle, |01\rangle, |10\rangle, |11\rangle\}$ is

$$U_{\mathrm{CNOT}} = I \oplus \sigma_x = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}. \tag{12}$$

In general, the qubits are denoted by horizontal lines in the quantum circuit diagrams and the gates as rectangles. The control nodes are marked by circles which

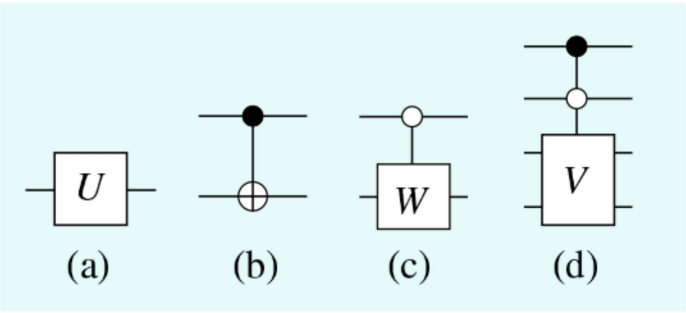

Figure 1: Quantum circuit symbols for (a) one-qubit gate, (b) CNOT, (c) controlled one-qubit gate, (d) twofold controlled two-qubit gate. In (c) the gate W acts only on the subspace in which the control qubit lies in the state $|0\rangle$ and in (d) the gate V operates on subspace in which the control qubits are in the state $|10\rangle$.

are connected to the associated gate by a vertical line. The effect of the control nodes is to limit the corresponding gates to act only on the subspace characterized by its control nodes. The nodes in the quantum circuit diagram can be black or white corresponding to the control qubit states $|1\rangle$ or $|0\rangle$, respectively (see Figs. 1(c) and (d)). Hereafter we refer to the k-fold controlled one-qubit gate V by $\mathrm{C}^k V$. When applied to an n qubit register, this gate operates in 2^{n-k}-dimensional target subspace consisting of those basis vectors for which the values of the control qubits match with those of the control nodes.

3 Uniformly controlled gates

3.1 Decomposition of uniformly controlled elementary rotations

Sequences of consequent controlled gates with slightly different control node configurations often appear in quantum circuit diagrams. Let us call a sequence of 2^k gates, each having a different sequence of k control nodes, a uniformly controlled U gate, see Fig. 2. The gate shown acts on the whole n-qubit register and, hence, it has $m = n - k$ target qubits denoted by the set T. Let us denote a gate of this kind by the symbol $F_T^k(U(2^m))$.

The concept of uniformly controlled gates with efficient gate implementation was for the first time introduced in Ref. [13] in the context of uniformly controlled rotations. It has also been utilized in decompositions of general n-qubit gates [13,15, 16,18], and in preparation of quantum states [15–17]. Bullock *et al.* have generalized uniformly controlled gates for a quantum register which is built of qudits, d-level ($d > 2$) quantum systems [19]. The methods to implement uniformly controlled z rotations are also closely related to the earlier work by Bullock and Markov [20].

Let us construct an elementary gate circuit for a uniformly controlled one-qubit

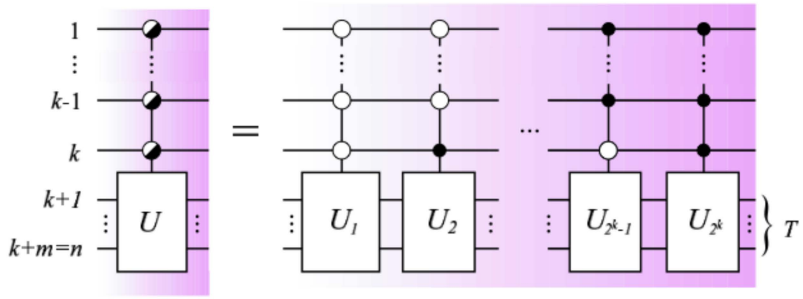

Figure 2: k-fold uniformly controlled m-qubit gate, $F_T^k\,(U(2^m))$, stands for a sequence of k-fold controlled gates U_i. Each of the gates acts on the set of target qubits T. Here $U_i \in U(2^m)$, where $i = 1, \ldots, 2^k$.

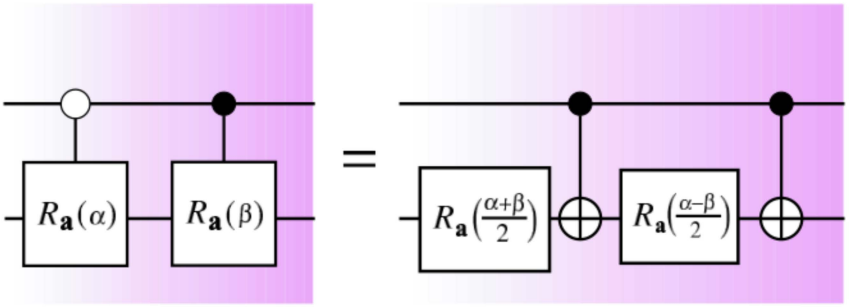

Figure 3: Implementation of a uniformly controlled one-parameter rotation $R_{\mathbf{a}}$ ($a_x = 0$) using the elementary gates.

gate. We present the decomposition of uniformly controlled one-parameter rotations, $F_t^k\,(R_{\mathbf{a}})$, separately since they require less gates to implement compared with general $F_t^k\,(U(2))$ gates. In a gate $F_t^k\,(R_{\mathbf{a}})$ the rotation angles vary, but the rotation axis is the same for each of the subrotations. In the spirit of Eq. (8), we may assume that the fixed axis $\mathbf{a}$ is perpendicular to x axis and, hence, we may employ Eq. (10) in the calculations.

Figure 3 shows how to decompose a gate $F_2^1\,(R_{\mathbf{a}})$ into two CNOTs and two elementary rotations. For the states with the the control qubit in state $|0\rangle$ the CNOTs are inactive and using Eq. (9) the rotation angles of the rotations $R_{\mathbf{a}}(\frac{\alpha+\beta}{2})$ and $R_{\mathbf{a}}(\frac{\alpha-\beta}{2})$ may be added to obtain the correct rotation $R_{\mathbf{a}}(\alpha)$. For control qubit sates $|1\rangle$ the rotation $R_{\mathbf{a}}(\frac{\alpha-\beta}{2})$ is negated according to Eq. (10) and the resulting gate is $R_{\mathbf{a}}(\beta)$. By adding qubits and control nodes we obtain the general step to eliminate control nodes from the uniformly controlled rotations as shown in Fig. 4(a).

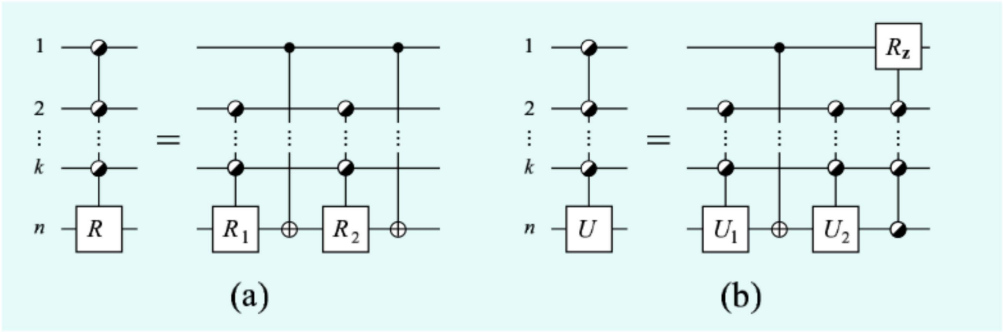

Figure 4: Decomposition of a uniformly controlled one-qubit gate. (a) One parameter rotation, (b) general one-qubit gate $U \in SU(2)$.

3.2 Decomposition of uniformly controlled one-qubit gates

To justify the control node elimination shown in Fig. 4(b), we need to introduce so-called constant quantum multiplexor. The idea is that a onefold uniformly controlled rotation is decomposed as

$$\begin{pmatrix} a & \\ & b \end{pmatrix} = \underbrace{\begin{pmatrix} r^\dagger & \\ & r \end{pmatrix}}_{R} \underbrace{\begin{pmatrix} u & \\ & u \end{pmatrix}}_{I\otimes u} \underbrace{\begin{pmatrix} d & \\ & d^\dagger \end{pmatrix}}_{D} \underbrace{\begin{pmatrix} v & \\ & v \end{pmatrix}}_{I\otimes v}, \tag{13}$$

where a, b, u and v are unitary and r and d are diagonal unitary 2×2 matrices. Here a and b are fixed by the uniformly controlled gate we are implementing, u and v correspond to the resulting one-qubit gates and the uniformly controlled z rotation corresponding to matrix r is to be tuned such that the diagonal matrix d separating the one qubit gates is independent of a and b.

Equation (13) yields the matrix equations

$$a = r^\dagger u d v, \tag{14}$$

$$b = r u d^\dagger v \tag{15}$$

or, equivalently,

$$X := a b^\dagger = r^\dagger u d^2 u^\dagger r^\dagger, \tag{16}$$

$$v = d u^\dagger r^\dagger b = d^\dagger u^\dagger r a. \tag{17}$$

Equation (16) may be recast into a form reminiscent of an eigenvalue decomposition:

$$rXr = u d^2 u^\dagger =: u \Lambda u^\dagger. \tag{18}$$

Note that X is fixed by the matrices a and b, but r can be chosen freely. By diagonalizing the matrix rXr, we find the similarity transformation u and the eigenvalue matrix $\Lambda = d^2$. The matrix v is obtained by inserting the results into Eq. (17).

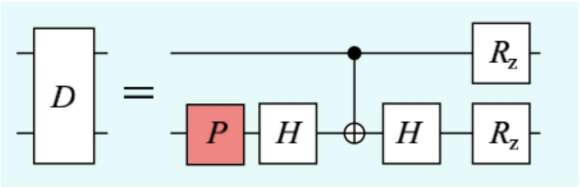

Figure 5: Elementary gate sequence for the D gate, where H is the Hadamard gate and $R_z = R_z(\pi/2)$. Gate $P = e^{-i\pi/4}$ is an adjustment of the global phase.

Since $X \in U(2)$, we may express it using the parametrization

$$X = \begin{pmatrix} x_1 & x_2 \\ -\bar{x}_2 & \bar{x}_1 \end{pmatrix} e^{i\phi/2}, \tag{19}$$

where $|x_1|^2 + |x_2|^2 = 1$ and $\det(X) = e^{i\phi}$. The characteristic polynomial of the matrix rXr is

$$\det(rXr - \lambda I) = \lambda^2 - \lambda \left(r_1^2 x_1 + r_2^2 \bar{x}_1\right) e^{i\phi/2} + r_1^2 r_2^2 e^{i\phi}. \tag{20}$$

Let us fix the freely tunable matrix r to be

$$r = \begin{pmatrix} e^{\frac{i}{2}\left[-\frac{\pi}{2} - \frac{\phi}{2} - \arg(x_1)\right]} & \\ & e^{\frac{i}{2}\left[\frac{\pi}{2} - \frac{\phi}{2} + \arg(x_1)\right]} \end{pmatrix}, \tag{21}$$

which implies the matrix d to be, indeed, independent of the matrices a and b. Namely

$$\Lambda = d^2 = \begin{pmatrix} e^{i\frac{\pi}{2}} & \\ & -e^{i\frac{\pi}{2}} \end{pmatrix}. \tag{22}$$

Hence, the diagonal multiplexing gate D obtains the fixed form $D = e^{i\frac{\pi}{4}\sigma_z \otimes \sigma_z}$, which can be realized straightforwardly using an Ising-type Hamiltonian or, alternatively, it can be decomposed into a CNOT and one-qubit gates as shown in Fig. 5.

The single qubit gates acting on the bottom qubit in Fig. 5 may be merged with the adjacent single qubit gates u and v resulting in single qubit gates u' and v' shown in Fig. 6, respectively. The z rotation acting on the top qubit in Fig. 5 may be correspondingly merged with the uniformly controlled z rotation in Fig. 6 and, hence, we have justified elimination of the control node for a onefold uniformly controlled one-qubit gate shown in Fig. 6. By adding qubits with control nodes we obtain the general step to eliminate control nodes from the UCGs as shown in Fig. 4(b).

We note that the uniformly controlled rotations in Fig. 4(a) have the same rotation axis and, hence, commute. Thus the first uniformly controlled rotation may be, as well, transfered to be the last gate. We call this procedure mirroring

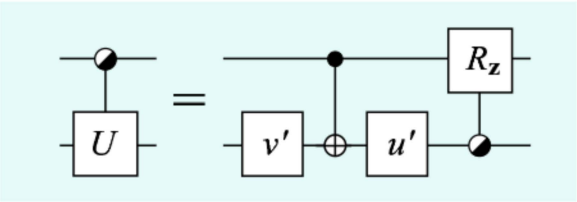

Figure 6: Constant quantum multiplexor for two qubits.

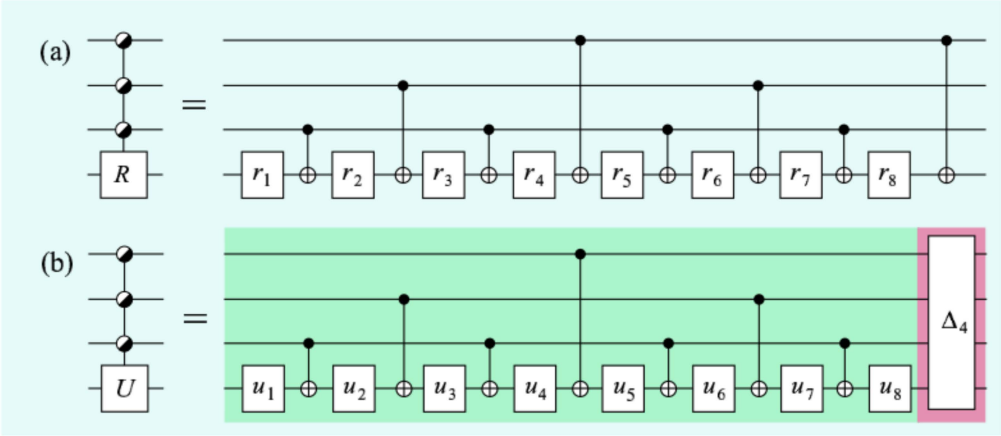

Figure 7: Quantum circuit realizing a threefold uniformly controlled (a) one-parameter rotation, (b) general one-qubit gate. In (a) $\{r_i\}$ stand for a one-parameter rotations and in (b) $\{u_i\}$ are general one-qubit gates. Here the gate Δ_4 corresponds to a diagonal 16×16 unitary matrix.

the circuit. By using the step of Fig. 4(a) recursively and mirroring every second outcome of the recursion we obtain the full decomposition of $F_t^k(R_\mathbf{a})$ using only 2^k one-qubit rotations $R_\mathbf{a}$ and the same number of CNOTs. An example of the case $k = 3$ is shown in Fig. 7(a). When decomposing general one-qubit UCGs, the step in Fig. 4(b) is to be used recursively. There we have to keep in mind that, actually, the CNOT may be taken to be the diagonal gate D show in Fig. 5. Hence, when the recursion is applied always on the leftmost UCG, all the resulting uniformly controlled z rotations may be merged with the adjacent UCGs except the rightmost ones which pile on to form a diagonal matrix Δ_4. The decomposition of $F_4^3(U(2))$ is shown in Fig. 7(b).

In general, the decomposition of a gate $F_t^k(U(2))$ includes an alternating sequence of 2^k one-qubit gates and $2^k - 1$ CNOTs which we denote by $\tilde{F}_t^k(U(2))$. Likewise, the implementation involves a cascade of k distinct uniformly controlled z rotations which corresponds to a single diagonal $(k+1)$-qubit gate Δ_{k+1}. However,

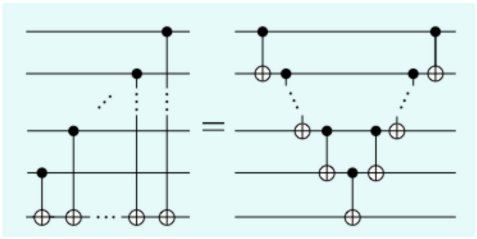

Figure 8: CNOT cascade which can be efficiently implemented using nearest-neighbor CNOTs [21].

the implementation of this part of the gate sequence can often be circumvented by merging it with the adjacent gates as shown in Sec. 5. In fact, if the qubit register is measured as such after the action of the gate $F_t^k(U(2))$, the diagonal gate may be left unimplemented since it does not change the probability amplitudes.

3.3 Nearest-neighbor decompositions

In the practical realization of a quantum computer, the spatial arrangement of qubits or other reasons may limit the interactions between the qubits. Let us consider a quantum register whose topology corresponds to that of a linear chain and which allows the gates to act only on nearest-neighbor qubits. This topology turns out to be amenable for implementing a uniformly controlled gate, which may have important consequences for experimentally realizing quantum computing.

The quantum circuit presented for a uniformly controlled gate can be translated efficiently into an array of nearest-neighbor gates. The technique is based on the circuit identity shown in Fig. 8. The strategy is to modify the decomposition shown in Fig. 4 by inserting an identity in the form of a CNOT cascade and its inverse, a similar cascade, into the circuit next to each CNOT. The inverse cascades are absorbed into the adjacent uniformly controlled gate. The remaining cascades, together with the original CNOTs, can be efficiently implemented using nearest-neighbor CNOTs as illustrated in Fig. 8. The control node elimination steps for the nearest-neighbor implementation are shown in Fig. 9.

The complexity of the nearest-neighbor implementation depends on the relative order of the target and control qubits, and the order in which the control qubits are eliminated. An efficient strategy is to first eliminate the control nodes that are furthest apart from the target. Furthermore, for the gates with numerous control nodes, it is advantageous to use a sequence of swap gates to move the target qubit next to the center of the chain before the operation and back after it. A swap gate can be realized using three consecutive CNOTs [2].

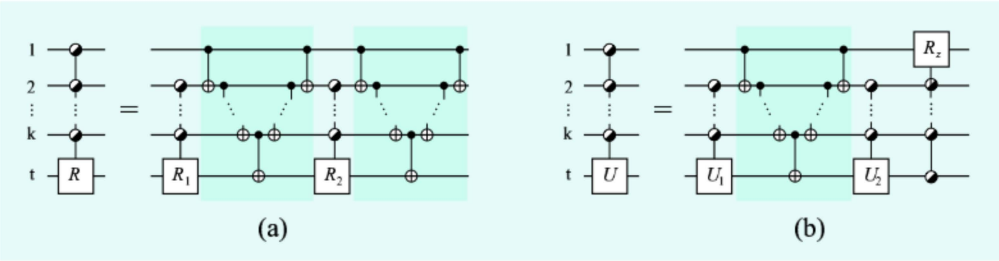

Figure 9: Method for reducing a uniformly controlled gate into nearest-neighbor gates: (a) uniformly controlled rotation and (b) general one-qubit gate. Here the circuit diagrams may also be mirrored horizontally.

Using this strategy a gate $\tilde{F}_t^{n-1}(U(2))$ can be implemented with at most

$$C_{U(2)}(n,s) = \frac{5}{6}2^n + 2n - 6s - \begin{cases} \frac{1}{3}, & n \text{ even} \\ \frac{5}{3}, & n \text{ odd} \end{cases} \tag{23}$$

nearest-neighbor CNOTs. Here $s = 1, \ldots, \lceil \frac{n}{2} \rceil$ is the distance of the target qubit t from the end of the chain. Figure 10(a) depicts the resulting circuit for the case $k = 4$ and $s = 1$. Similar treatment for gate $F_t^{n-1}(R_{\mathbf{a}})$ yields a quantum gate array with

$$C_R(n,s) = \frac{5}{6}2^n + 3n - 6s - \begin{cases} \frac{4}{3}, & n \text{ even} \\ \frac{5}{3}, & n \text{ odd} \end{cases} \tag{24}$$

nearest-neighbor CNOTs. Figure 10(b) displays an example circuit for the case $k = 4$ and $s = 1$.

We note that the uniformly controlled one-qubit gate carries $3 \cdot 2^k$ degrees of freedom, and requires roughly the same number of elementary gates for its implementation. Thus an array of nearest-neighbor CNOTs provides an efficient implementation for uniformly controlled one-qubit gates, and therefore for any uniformly controlled gate. In particular this can be utilized to efficiently implement unstructured unitary transformations. Furthermore, the structure of the nearest-neighbor circuit allows several gate operations to be executed in parallel which may further reduce the execution time of the algorithm.

4 QR decomposition

Numerical matrix computation [12] is a field of mathematics that provides useful tools to construct and manipulate quantum gate arrays. For example, the theorem of QR decomposition states that for each complex matrix A there exists a unitary matrix Q and an upper triangular matrix R such that $A = QR$. Here the matrix Q may be a product two-level matrices called Givens rotations [22]. For unitary matrices A, the resulting matrix R is essentially an identity. Thus the sequence of

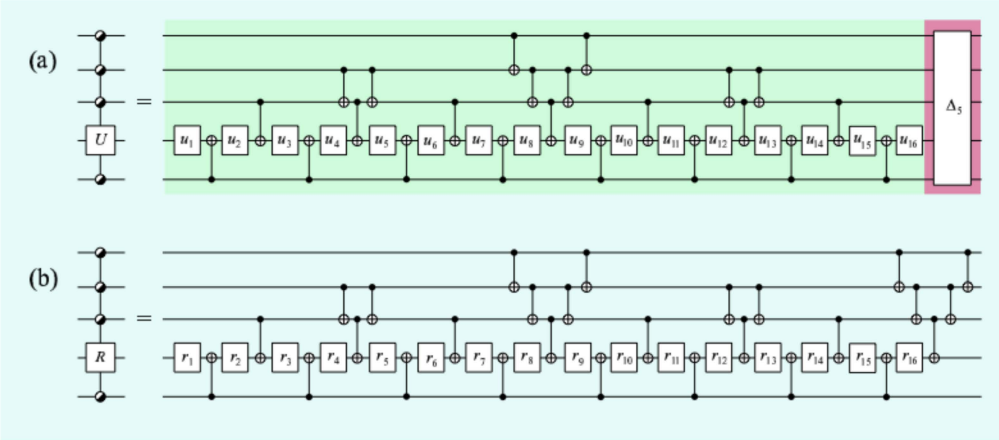

Figure 10: Implementation of threefold uniformly controlled (a) general one-qubit gate and (b) one-parameter rotation. Here gates $\{r_i\}$ are generic rotations about x axis and gates $\{u_i\}$ belong to $SU(2)$. The alternating sequence of CNOTs and u_i gates is denoted by $\tilde{F}_5^4\,(U(2))$. The rightmost sequence of uniformly controlled z rotations corresponds to a single diagonal gate, denoted by Δ_5.

Givens rotations yields a decomposition of any unitary matrix into two-level matrices. Consequently, these two-level matrices may be decomposed into elementary gates as shown below. Traditionally, a technique based on this principle is employed in quantum computation to find the elementary decomposition of an unstructured unitary matrix [6, 9, 23]. Reference [11] presents improvements to the traditional construction that eventually lead to the quantum gate decomposition of minimal complexity $O(4^n)$.

Let us outline how to find the sequence of Givens rotations, the product of whom implements any unitary matrix $U \in SU(2^n)$. In the case $n = 1$, a Givens rotation $G \in SU(2)$ corresponding to a vector $\mathbf{b} = (b_1\, b_2)^T$ may be defined as

$$G\,\mathbf{b} = \frac{1}{\sqrt{|b_1|^2 + |b_2|^2}} \begin{pmatrix} b_1^* & b_2^* \\ -b_2 & b_1 \end{pmatrix} \begin{pmatrix} b_1 \\ b_2 \end{pmatrix} = \begin{pmatrix} \sqrt{|b_1|^2 + |b_2|^2} \\ 0 \end{pmatrix}. \tag{25}$$

For general number of qubits n, a Givens rotation is a two-level matrix acting non-trivially only on a subspace spanned by two basis vectors, for example, $|e_j\rangle$ and $|e_k\rangle$. When a Givens rotation is used to nullify elements of a matrix $U \in SU(N)$ we also need to specify the column which is used as the vector corresponding to the rotation. Hence, we define a Givens rotation ${}^iG_{j,k}$ to be a two-level complex matrix which selectively nullifies the element on the i^{th} column and the j^{th} row of

the matrix U against the element on the i^{th} column and the k^{th} row. For example

$$ {}^{1}G_{N,N-1}U = \begin{pmatrix} u_{1,1} & u_{1,2} & \cdots & u_{1,N} \\ \vdots & \vdots & \ddots & \vdots \\ u_{N-2,1} & u_{N-2,2} & \cdots & u_{N-2,N} \\ \tilde{u}_{N-1,1} & \tilde{u}_{N-1,2} & \cdots & \tilde{u}_{N-1,N} \\ 0 & \tilde{u}_{N,2} & \cdots & \tilde{u}_{N,N} \end{pmatrix}, \tag{26} $$

where the elements of $\tilde{U}$ that differ from those of U are indicated with the tilde.

Applying ${}^{1}G_{N-1,N-2}$ to the modified matrix $\tilde{U}$ we can nullify the element $\tilde{u}_{N-1,1}$ and similarly the whole first column, except the diagonal element. The unitarity of the matrix U fixes its absolute value to unity and the definition of a Givens rotation in Eq. (25) assures that the phase of the diagonal element vanishes, *i.e.*, it obtains value 1. The further application of the method to the columns from 2 to $N-1$ results in an identity matrix as

$$ \left(\prod_{i=1}^{2^n-1} \prod_{j=i+1}^{2^n} {}^{2^n-i}G_{j,j-1} \right) U = I, \tag{27} $$

where the product of the non-commuting matrices is taken from left to right as always in this chapter. Equation (27) yields the factorization of the arbitrary matrix $U \in SU(2^n)$ using Givens rotations

$$ U = \left(\prod_{i-1}^{2^n-1} \prod_{j-1}^{2^n-i} {}^{i}G^{\dagger}_{2^n-j+1,2^n-j} \right), \tag{28} $$

which introduces an implementation of an arbitrary quantum gate provided that an elementary gate presentation of each of the Givens rotations is known. We note the non-zero off-diagonal elements of ${}^{i}G_{j,k}$ by 2×2-matrix ${}^{i}\Gamma_{j,k}$.

In the first presentation of the QR decomposition for arbitrary quantum gates [6], the basis vectors were ordered using standard binary coding. Thus the Givens rotations acting on adjacent basis vectors do not directly correspond to any known gate. However, if the basis vectors are permuted before the action of every rotation and permuted back after the action, the rotations may be written as fully controlled one-qubit gates. The permutation for each $O(4^n)$ rotations needed of the order of n fully controlled NOT gates each of which required of the order of n^2 CNOTs. Hence, the complexity of the whole decomposition turned out to be $O(n^3 4^n)$.

Instead of labelling the basis vectors using standard binary coding, the binary reflected Gray code was employed in Ref. [11]. The special property of any Gray code ordered basis is that only one bit changes between the adjacent basis vectors $|e_i\rangle$ and $|e_{i+1}\rangle$, see Fig. 11(a). The important consequence of this is that the operations limited to the subspace spanned by $|e_i\rangle$ and $|e_{i+1}\rangle$ take the form of a $\mathrm{C}^{n-1}V$ gate,

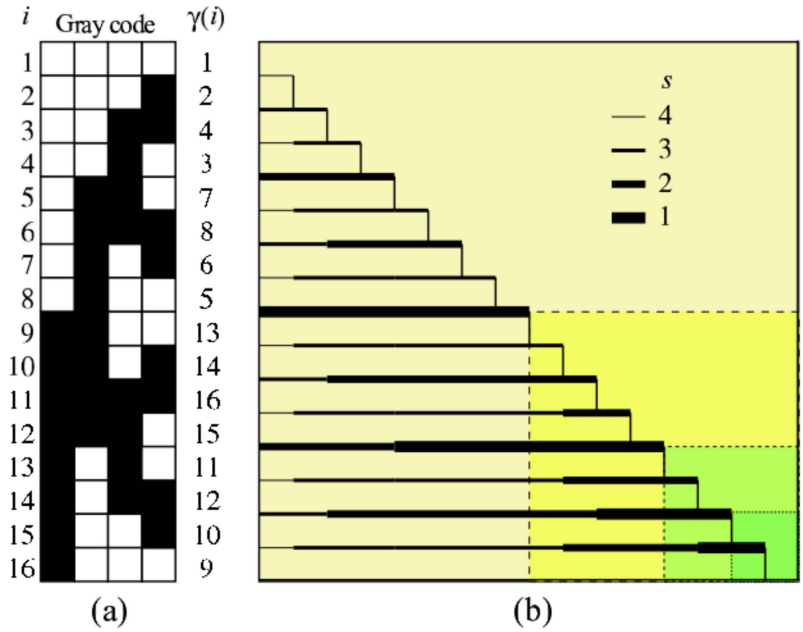

Figure 11: (a) Four bit Gray code. White squares stand for bit values 0 and black squares denote 1. (b) The number of control nodes needed for the Givens rotation nullifying the elements of the matrix U. The width of the line s between the matrix elements represents the number of control nodes which may be eliminated.

where $V \in SU(2)$. Consequently, each of the $2^{n-1}(2^n-1)$ Givens rotations $\{{}^iG_{j,j-1}\}$ can be implemented using only one $\mathrm{C}^{n-1}V$ gate and no basis permuting gates are needed between them. Since a $\mathrm{C}^{n-1}V$ gate may be decomposed into $O(n)$ CNOTs [6], the decomposition has a complexity $O(n4^n)$ at this point. We note that actually we may, as well, label the basis vectors using the standard binary coding but, instead, the order in which the elements of the matrix $U \in SU(2^n)$ are nullified must be chosen such that the Givens rotations operate non-trivially only to basis vectors with binary presentations differing only in one bit. Provided that the basis vectors are labelled using the standard binary coding, the matrices ${}^{2^n-i}G_{j,j-1}$ in Eq. (27) become ${}^{\gamma(2^n-i)}G_{\gamma(j),\gamma(j-1)}$, where the function $\gamma(i)$ gives the integer value of the i^{th} element in the binary reflected Gray code, see Fig. 11.

Furthermore, we find that only a small fraction of the control nodes in the fully controlled one-qubit gates appears to be essential for the final result of the decomposition. If s control nodes are removed from a $\mathrm{C}^{n-1}({}^i\Gamma_{j,j-1})$ gate, the matrix representation ${}^iG_j^s$ of such an operation is no more two-level, but rather 2^{s+1}-level, *i.e.*, the matrix ${}^iG_j^s$ operates with the matrix ${}^i\Gamma_{j,j-1}$ to all pairs of basis vectors which satisfy the remaining control conditions. Once some element of the matrix U we are decomposing becomes zero in the diagonalization process, we must remove

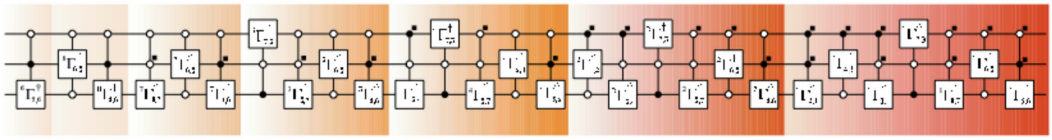

Figure 12: Quantum circuit equivalent to an arbitrary three-qubit quantum gate $U \in SU(8)$. The control nodes indicated with a black square on the upper right hand side corner are superfluous and may be omitted to decrease the complexity of the decomposition.

control nodes from the following fully controlled gates in such a way that the zeroed element does not mix with the non-zero elements.

Figure 11(b) illustrates the determination of the number of control nodes necessary in the diagonalization. The total number of gates in the implementation depends on the number of the control nodes in each of the involved gates. Let us denote by $g_n(k)$ the number of $\mathrm{C}^k V$ gates requires for the whole diagonalization process of an n-qubit gate. In Ref. [11], a recursion relation for $g_n(k)$ was derived. The relation has an awkward analytic solution and, therefore, it was estimated from above as

$$g_n(n-i) \leq 2^{n+i}. \tag{29}$$

Equation (29) shows that the number of k-fold controlled gates decreases exponentially with the number of control nodes. On the other hand, gate $\mathrm{C}^k V$ takes $O(n)$ gates to implement [6]. These results together imply that the gate array for an n-qubit unitary gate involves $O(4^n)$ elementary gates. Figure 12 shows an example of the quantum circuit equivalent to an arbitrary three-qubit quantum gate $U \in SU(8)$.

To calculate the number of elementary gates, we use the decompositions described in Ref. [6]. For large n, the leading contribution to the number of CNOTs is approximately 8.7×4^n, while the upper bound from Eq. (29) yields approximately 11×4^n. We note that neither one of the two techniques alone, the Gray code ordered basis vectors nor the elimination of the control nodes suffices to decrease the circuit complexity to $O(4^n)$. As a curiosity, the technique to eliminate control nodes has recently been generalized and adopted again to the numerical matrix computation [24].

5 Cosine-sine decomposition

5.1 Recursive cosine-sine decomposition

The CSD of a unitary $2^n \times 2^n$ matrix may be expressed as [25]

$$U = \underbrace{\begin{pmatrix} u_1 & 0 \\ 0 & u_2 \end{pmatrix}}_{U_1} \underbrace{\begin{pmatrix} c & s \\ -s & c \end{pmatrix}}_{A} \underbrace{\begin{pmatrix} u_3 & 0 \\ 0 & u_4 \end{pmatrix}}_{U_2}, \tag{30}$$

where $\{u_k\}$ are unitary $2^{n-1} \times 2^{n-1}$ matrices and the real diagonal matrices c and s are of the form $c = \mathrm{diag}_l(\cos\theta_l)$ and $s = \mathrm{diag}_l(\sin\theta_l)$ $(l = 1, \ldots, 2^{n-1})$. The matrix A corresponds to a uniformly controlled y rotation $F_1^{n-1}(R_y)$ with rotation angles $\{\theta_l\}$ and the matrices U_1 and U_2 to uniformly controlled $(n-1)$-qubit gates $F_T^1\left(SU(2^{n-1})\right)$. By applying Eq. (30) recursively to the uniformly controlled multi-qubit gates until we only have uniformly controlled one-qubit gates, we obtain a decomposition

$$U(2^n) = F_n^{n-1}\left(U(2)\right) \prod_{i=1}^{2^{n-1}-1} F_{n-\zeta(i)}^{n-1}\left(R_y\right) F_n^{n-1}\left(U(2)\right), \tag{31}$$

where ζ is the ruler function [26].

We begin to decompose the rightmost gate in Eq. (33) into elementary gates by writing an identity $I = \Delta_n \Delta_n^*$ between the gates $F_{n-\zeta(2^{n-1}-1)}^{n-1}(R_y)$ and $F_n^{n-1}(U(2))$. Here we choose Δ such that

$$F_n^{n-1}\left(U(2)\right) = \Delta_n \tilde{F}_n^{n-1}\left(U(2)\right), \tag{32}$$

where the gate $\tilde{F}_n^{n-1}(U(2))$ introduced in Sec. 3.2 needs only $2^{n-1}-1$ CNOTs to implement. We are now left with the product

$$\begin{aligned} U(2^n) \;=\; & F_n^{n-1}\left(U(2)\right) \left[\prod_{i=1}^{2^{n-1}-2} F_{n-\zeta(i)}^{n-1}\left(R_y\right) F_n^{n-1}\left(U(2)\right)\right] && (33) \\ & \times F_{n-\zeta(2^{n-1}-1))}^{n-1}\left(R_y\right) \Delta_n \tilde{F}_n^{n-1}\left(U(2)\right), && (34) \end{aligned}$$

where the product $F_{n-\zeta(2^{n-1}-1))}^{n-1}(R_y)\,\Delta_n$ may be written as a single uniformly controlled one-qubit gate $F_{n-\zeta(2^{n-1}-1))}^{n-1}(U(2))$. Continuing to change the $F^{n-1}(U(2))$ gates into $\tilde{F}^{n-1}(U(2))$ gates by adding diagonal gates, we finally obtain the decomposition

$$U(2^n) = \Delta_n \tilde{F}_n^{n-1}\left(U(2)\right) \prod_{i=1}^{2^{n-1}-1} \tilde{F}_{n-\gamma(i)}^{n-1}\left(U(2)\right) \tilde{F}_n^{n-1}\left(U(2)\right). \tag{35}$$

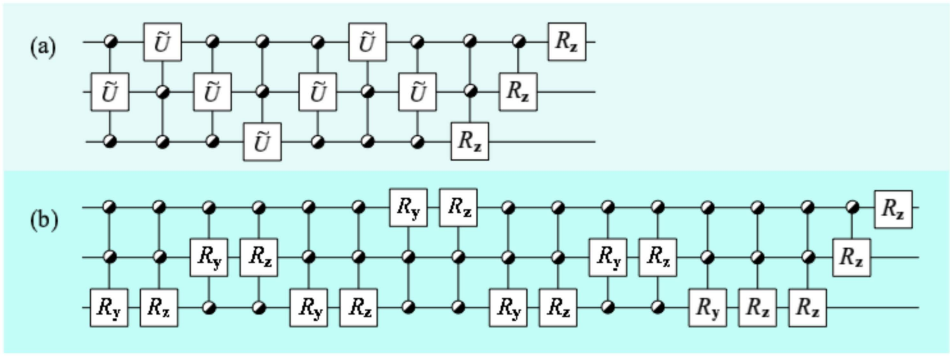

Figure 13: Quantum circuit for a three-qubit gate obtained using (a) the current CSD and (b) an alternative CSD.

There are $2^n - 1$ $\tilde{F}^{n-1}(U(2))$ gates in Eq. (35), each of which may be decomposed into $2^{n-1} - 1$ CNOTs. In addition, we have to implement the diagonal gate Δ_n using $2^n - 2$ CNOTs [20]. Actually, one more CNOT may be eliminated [15] and, hence, the current CSD requires $4^n/2 - 2^{n-1} - 2$ CNOTs. The circuit diagram for the CSD in the case $n = 3$ is shown in Fig. 13(a), where the diagonal gate Δ_3 is written as a cascade of uniformly controlled z rotations [13]. There exists also a slightly different version of the CSD where the matrix $U \in SU(2^n)$ is decomposed only into uniformly controlled z and y rotations [13]. An example of this alternative CSD in shown in Fig. 13(b). Actually, the alternative CSD is obtained also from the current one by writing the rightmost UCG in the product of Eq. (33) as a product $F^{n-1}(R_z)\,F^{n-1}(R_y)\,F^{n-1}(R_z)$. Being diagonal, the gate $F^{n-1}(R_z)$ may be merged into the adjacent UCG and the process can be continued until the last $F^{n-1}(R_z)$ arising from the leftmost UCG may be merged to the diagonal gate Δ_n.

5.2 Top down approach

In addition to the CSD described above, Ref. [16] presents an alternative approach employing the cosine-sine decomposition. This method is called NQ decomposition[3] and it is almost as efficient as the CSD discussed in Sec. 5.1. The first step of the NQ method is the same as in the CSD shown in Fig. 14(a), see also Eq. (30). However, the CSD step is not used recursively but, instead, the control nodes in the UCG are eliminated using quantum multiplexor shown in Fig. 14(b). After the application of the CSD and the quantum multiplexor, we are left with three uniformly controlled rotations separating four uncontrolled $n-1$ qubit gates. Since the NQ step produces pure gates acting on fewer qubits it is also called a top down approach.

Let us now motivate the validity of the quantum multiplexor. It is very similar to the constant quantum multiplexor presented in Sec. 3.2 but the matrices corre-

[3]NQ stands for n qubits.

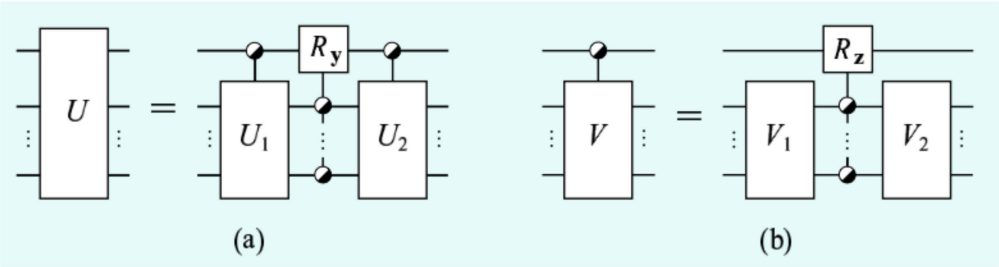

Figure 14: Circuit diagram for (a) cosine-sine decomposition and (b) Quantum multiplexor.

sponding to Eq. (13) are $2^{n-1} \times 2^{n-1}$-dimensional, the matrix r is omitted and the diagonal matrix d may depend and the matrices a and b. Actually, it is an open problem whether there exists a constant quantum multiplexor in this general case, *i.e.*, can we find diagonal $r \in SU(2^n)$ for every $X \in SU(2^n)$ such that the eigenvalues of rXr are fixed. The matrix equation corresponding to Fig. 14(b) reads as

$$\begin{pmatrix} a & \\ & b \end{pmatrix} = \begin{pmatrix} u & \\ & u \end{pmatrix} \begin{pmatrix} d & \\ & d^\dagger \end{pmatrix} \begin{pmatrix} v & \\ & v \end{pmatrix}, \tag{36}$$

where a, b, u and v are unitary and d is diagonal unitary $2^{n-1} \times 2^{n-1}$ matrices. Equation (36) yields the matrix equations

$$a = udv, \tag{37}$$

$$b = ud^\dagger v \tag{38}$$

or, equivalently,

$$ab^\dagger = ud^2u^\dagger, \tag{39}$$

$$v = du^\dagger b = d^\dagger u^\dagger a. \tag{40}$$

By diagonalizing the matrix $ab^\dagger$, we find the similarity transformation u and the eigenvalue matrix d^2. The matrix v is obtained by inserting the results into Eq. (40). Hence, we have proven the quantum multiplexor in Fig. 14(b).

The NQ step is continued recursively to all the gates except the uniformly controlled rotations until the two-qubit level is encountered. The two-qubit gates are decomposed using the minimal elementary gate construction shown in Fig. 15. In fact, diagonal gates commute with the control nodes of the UCG and, hence, all but one of the resulting two-qubit gates may be implemented up to diagonal, *i.e.*, using only two CNOTs as shown in the leftmost part of Fig. 15.

We will now calculate the number of the CNOTs involved in to NQ decomposition of an unstructured $U \in SU(2^n)$. Let us denote this number by a_n. Since the NQ

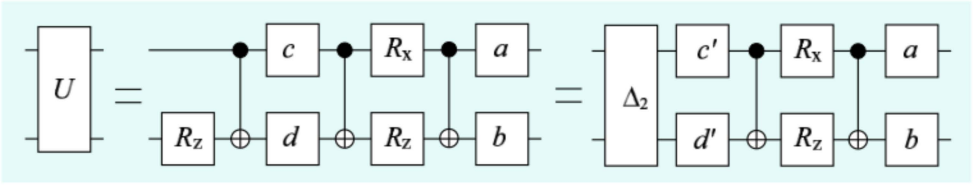

Figure 15: The minimal elementary gate construction for a two-qubit gate [10].

step produces four unstructured gates in $SU(2^{n-1})$ and three $F^{n-1}(R)$ gates each of which may be implemented using 2^{n-1} CNOTs, we obtain a recursion relation

$$a_n = 4a_{n-1} + \frac{3}{2}2^n. \tag{41}$$

Using the above discussed condition $a_2 = 2$ and adding one CNOT from the only two-qubit gate which needs three CNOTs for its implementation we obtain the result

$$a_n = \frac{1}{2}4^n - \frac{3}{2}2^n + 1. \tag{42}$$

Thus compared with the number of CNOTs from the CSD $\frac{1}{2}4^n - \frac{1}{2}2^n - 2$, the NQ decomposition yields the same result in the leading order. However, when we compare the number on one-qubit gates or alternatively elementary rotations, the CSD is found to be more efficient, see Table. 1.

Gate type	NQ	CSD
CNOT	$\frac{1}{2}4^n - \frac{3}{2}2^n + 1$	$\frac{1}{2}4^n - \frac{1}{2}2^n - 2$
R_y, R_z	$\frac{9}{8}4^n - \frac{3}{2}2^n + 3$	$4^n - 1$
or $SU(2)$	$\frac{17}{24}4^n - \frac{3}{2}2^n - \frac{1}{3}$	$\frac{1}{2}4^n + \frac{1}{2}2^n - n - 1$

Table 1: Comparison of the gate counts required to implement a general n-qubit gate using the NQ decomposition [16] and the recursive CSD for unstructured n-qubit gates.

Actually, the number of CNOTs in the NQ decomposition may be reduced by noting that the resulting uniformly controlled y rotations may be always implemented up to a diagonal gate as seen from Fig. 14(a). Since $\tilde{F}^k(U(2))$ gate needs one CNOT less to implement than $F^k(R_y)$ we obtain a recursion relation

$$a_n = 4a_{n-1} + \frac{3}{2}2^n - 1, \tag{43}$$

the solution of which is found to be

$$a_n = \frac{23}{48}4^n - \frac{3}{2}2^n + \frac{1}{3} \tag{44}$$

This result is the first known to require less than $\frac{1}{2}4^n$ CNOTs in the leading order.

6 Local state preparation

The execution of any quantum algorithm requires a certain initial state as an input. Depending on the physical realization of the quantum computer, convenient initialization procedures may only produce a limited range of states possibly not containing the desired initial state. This brings up the problem of local state preparation[4], *i.e.*, how to implement the transformation of an arbitrary quantum state into another one.

The configuration space of the n-qubit quantum register is 2^n-dimensional complex space. Excluding the global phase and state normalization, we find that the general unitary transformation transforming a given n-qubit state into another must have at least $2 \times 2^n - 2$ real degrees of freedom. Hence, in the worst-case scenario, the corresponding quantum circuit should involve at least $2^{n+1} - 2$ elementary rotations, each carrying one degree of freedom. Since each of the CNOTs can bind at most four elementary rotations [10], at least $\lceil \frac{1}{4}(2^{n+1} - 3n - 2) \rceil$ of them are needed. However, no quantum circuit construction embodying the minimal complexity has been presented in the literature. An upper bound for the number of gates needed for state preparation has been considered by Knill [8], who found that no more than $O(n2^n)$ gates provide the circuit implementing the transformation. More recently, a sufficient circuit of $O(2^n)$ elementary gates was obtained in Ref. [27] (see also Ref. [17]) as a special case of the method developed for QR decomposition of a general quantum gate in Ref. [11]. In this section, we present the best known method to execute the local state preparation first introduced in Ref. [15].

Our aim is to build a fixed structure circuit which takes any given input state $|a\rangle_n$ to any chosen state $|b\rangle_n$. We begin by noting that once we know an efficient circuit taking $|a\rangle_n$ to any fixed vector, for example $|e_1\rangle$, we may use the inverse that circuit with different parameters to transform $|e_1\rangle$ to $|b\rangle_n$. The $|a\rangle_n$ to $|e_1\rangle_n$ transformation consists of a sequence of gate pairs

$$S_a = \prod_{i=1}^{n} \left[\left(F_i^{i-1}(R_y) \, F_i^{i-1}(R_z) \right) \otimes I_{2^{n-i}} \right]. \tag{45}$$

The effect of the gate pair $F_i^{i-1}(R_y) \, F_i^{i-1}(R_z)$ on the state $|a\rangle_i$ is to nullify half of its elements:

$$F_i^{i-1}(R_y) \, F_i^{i-1}(R_z) \, |a\rangle_i = |a'\rangle_{i-1} \otimes |0\rangle_1. \tag{46}$$

Hence, each successive gate pair nullifies half of the elements of the state vector that have not yet been zeroed, and we have $S_a|a\rangle_n = |e_1\rangle_n$ up to a global phase.

Now we note that the pair of gates $F_n^{n-1}(R_y) \, F_n^{n-1}(R_z) = F_n^{n-1}(U(2))$ may be replaced by the gate

$$\tilde{F}_n^{n-1}(U(2)) = \Delta_n^{\dagger} F_n^{n-1}(U(2)), \tag{47}$$

[4]We use the word local to separate the state preparation discussed here from the remote state preparation related to quantum teleportation.

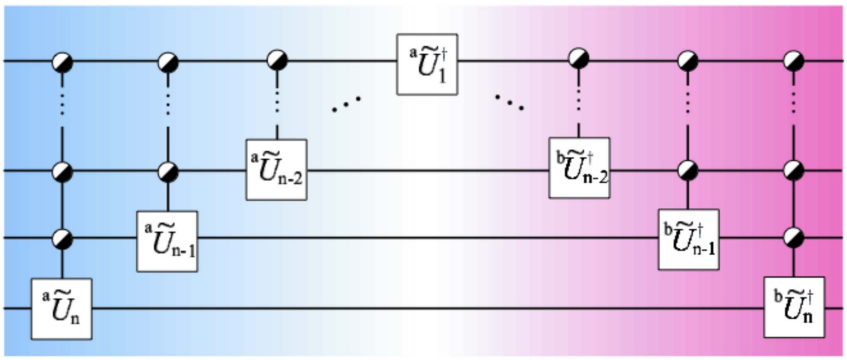

Figure 16: Quantum circuit for transforming an arbitrary n-qubit state vector $|a\rangle_n$ into desired state vector $|b\rangle_n$. The resulting gates are of the form $\tilde{F}_i^{i-1}(U(2))$ which is efficient to implement, see Fig. 7.

since the diagonal gate

$$\Delta_n^\dagger = \Delta_{n-1}^{0\,\dagger} \otimes |0\rangle\langle 0| + \Delta_{n-1}^{1\,\dagger} \otimes |1\rangle\langle 1| \tag{48}$$

does not mix the states;

$$\begin{aligned} \Delta_n^\dagger F_n^{n-1}(U(2))\,|a\rangle_n &= \Delta_n^\dagger \left(|a'\rangle_{n-1} \otimes |0\rangle_1\right) \\ &= \left(\Delta_{n-1}^{0\,\dagger}|a'\rangle_{n-1}\right) \otimes |0\rangle_1 \\ &= |a''\rangle_{n-1} \otimes |0\rangle_1. \end{aligned} \tag{49}$$

After combining $n-1$ pairs of adjacent $F_{k+1}^k(R_y)\,F_{k+1}^k(R_z)$ gates where $k = 1, ..., n-1$ we find that the entire circuit for transforming $|a\rangle$ to $|b\rangle$ requires $2 \cdot 2^n - 2n - 2$ CNOTs and $2 \cdot 2^n - n - 2$ one-qubit gates. If $|a\rangle$ or $|b\rangle$ coincides with one of the basis vectors $|e_i\rangle$, the gate counts are halved in the leading order. Figure 16 showns the circuit diagram of the whole local state preparation $S_b^\dagger S_a$.

7 Conclusion

In this chapter we have studied efficient implementations of general n-qubit gates within the quantum circuit model. From the two philosophically different approaches, the cosine-sine decomposition based methods were found to lead to smaller gate counts than the QR decomposition based ones. The QR decomposition, the CSD and the NQ decomposition are compared in the required number of CNOTs and the total number of elementary gates in Tables. 2 and 3, respectively. The QR decomposition is observed to have clearly the highest gate counts. The CSD requires slightly more CNOTs compared with the NQ decomposition but, on the other hand, the total number of elementary gates is noticeably larger in the NQ decomposition.

Table 2: Comparison of the number of CNOTs needed in different decompositions of general n-qubit gates.

n	1	2	3	4	5	6	7	8	9
QR	0	4	64	536	4156	22618	108760	486052	2078668
CSD	0	4	26	118	494	2014	8126	32638	130814
NQ	0	3	21	105	465	1953	8001	32385	130305

Table 3: Comparison of the total number of gates needed in different decompositions of general n-qubit gates.

n	1	2	3	4	5	6	7	8	9
QR	1	14	136	980	7384	42390	208820	944280	4062520
CSD	1	11	58	249	1016	4087	16374	65525	262132
NQ	1	10	54	262	1142	4758	19414	78422	315222

A special class of gates, called uniformly controlled gates, was introduced as basic building blocks of quantum circuits. In fact, the power of the gate-efficient methods employing the cosine-sine decomposition lies deep on the efficient implementation of uniformly controlled rotations and two-qubit gates. These gates also proved to be essential in a circuit transforming an arbitrary quantum state into another, *i.e.*, performing local state preparation. In the case of a one-dimensional chain of qubits, the uniformly controlled one-qubit gates were decomposed using only nearest-neighbor gates, which may turn to be essential for the experimental realizations of quantum computers. By cleverly using the nearest-neighbor decomposition in the recursive CSD of an n-qubit gate, it has been shown [15] that only $\frac{5}{6}4^n$ CNOTs are needed in the leading order. It is quite surprising that the gate count is increased by a factor of less than two, if the restriction to nearest-neighbor interactions is added.

In conclusion, we have reviewed the development of the circuit constructions of arbitrary quantum gates, slightly improved the lowest known gate count for the CNOTs to $\frac{23}{48}4^n$ in the leading order, discussed the local state preparation, and the circuits employing only nearest-neighbor CNOTs.

References

[1] Ballentine, L. E. *Quantum Mechanics: a Modern Development*; World Scientific, Singapore, 1998.

[2] Nielsen, M. A.; Chuang, I. L. *Quantum Computation and Quantum Information*; Cambridge University Press: Cambridge, 2000.

[3] Shor, P. W. *IEEE Proc. 35nd Annual Symposium on Foundations of Computer Science* 1994, 124.

[4] Gisin, N.; Ribordy, G.; Tittel, W.; Zbinden, H. *Rev. Mod. Phys.* 2002, vol. 74, 145.

[5] Deutsch, D. *Proc. R. Soc. of Lond. A* 1989, vol. 425, 73.

[6] Barenco, A.; Bennett, C. H.; Cleve, R.; DiVincenzo, D. P.; Margolus, N. H.; Shor, P. W.; Sleator, T.; Smolin, J. A.; Weinfurter, H. *Phys. Rev. A* 1995, vol. 52, 3457.

[7] Lloyd, S. Phys. *Rev. Lett.* 1995, vol. 75, 346.

[8] Knill, E.; quant-ph/9508006, 1995.

[9] Cybenko, G. *Computing in Science and Engineering* 2001, vol. 3, 27.

[10] Shende, V. V.; Markov, I. L.; Bullock, S. S. *Phys. Rev. A* 2004, vol. 69, 062321.

[11] Vartiainen, J. J.; Möttönen, M.; Salomaa, M. M. *Phys. Rev. Lett.* 2004, vol. 92, 177902.

[12] Golub, G. H.; Van Loan, C. F. *Matrix Computations* 3rd ed.; Johns Hopkins Press: Baltimore, 1996.

[13] Möttönen, M.; Vartiainen, J. J.; Bergholm, V.; Salomaa, M. M. *Phys. Rev. Lett.* 2004, vol. 93, 130502.

[14] Tucci, R. R.; quant-ph/9902062, 1999.

[15] Bergholm, V.; Vartiainen, J. J.; Möttönen, M.; Salomaa, M. M.; quant-ph/0410066, 2004.

[16] Shende, V. V.; Bullock, S. S.; Markov, I. L.; quant-ph/0406176.

[17] Möttönen, M.; Vartiainen, J. J.; Bergholm, V.; Salomaa, M. M.; quant-ph/0407010, 2004.

[18] Tucci, R. R.; quant-ph/0411027, 2004.

[19] Bullock, S. S.; Brennen, G. K.; O'Leary, D. P.; quant-ph/0410116, 2004.

[20] Bullock, S. S.; Markov, I. L. *Quant. Inf. and Comp.* 2004, vol. 4, 27.

[21] Tucci, R. R.; quant-ph/0407215, 2004.

[22] Givens, W. J. *Soc. Ind. Appl. Math* 1958, vol. 6, 26.

[23] Aho, A. V.; Svore, K. M.; quant-ph/0322008, 2003.

[24] O'Leary, D. P.; Bullock, S. S.; unpublished, 2004.

[25] Paige, C. C.; Wei, M. *Linear Algebra and Appl.* 1994, vol. 208, 303.

[26] Guy, R. K., *Unsolved Problems in Number Theory, 2nd ed.*; Springer-Verlag: New York, 1994; p. 224.

[27] Shende, V. V.; Markov, I. L.; quant-ph/0401162, 2004.