Signal processing techniques for efficient compilation of controlled rotations in trapped ions

Groenland, K.; Witteveen, F.; Schoutens, K.; Gerritsma, R.

Published in:
New Journal of Physics

DOI:
10.1088/1367-2630/ab8830

Citation for published version (APA):
Groenland, K., Witteveen, F., Schoutens, K., & Gerritsma, R. (2020). Signal processing techniques for efficient compilation of controlled rotations in trapped ions. New Journal of Physics, 22(6), [063006]. https://doi.org/10.1088/1367-2630/ab8830
Signal processing techniques for efficient compilation of controlled rotations in trapped ions

To cite this article: Koen Groenland et al 2020 New J. Phys. 22 063006

View the article online for updates and enhancements.
Signal processing techniques for efficient compilation of controlled rotations in trapped ions

Koen Groenland, Freek Witteveen, Kareljan Schoutens and Rene Gerritsma

1 QuSoft and Centrum Wiskunde en Informatica (CWI), Science Park 123, 1098 XG Amsterdam, The Netherlands
2 Institute of Physics, University of Amsterdam, Science Park 904, 1098 XH Amsterdam, The Netherlands
3 Korteweg-de Vries Institute for Mathematics, University of Amsterdam, Science Park 105-107, 1098 XG Amsterdam, The Netherlands
4 Author to whom any correspondence should be addressed.

E-mail: koen.groenland@gmail.com

Keywords: Quantum Computing, Trapped ions, Quantum signal processing, Quantum gates

Abstract

Quantum logic gates with many control qubits are essential in many quantum algorithms, but remain challenging to perform in current experiments. Trapped ion quantum computers natively feature the Mølmer–Sørensen (MS) entangling operation, which effectively applies an Ising interaction to all pairs of qubits at the same time. We consider a sequence of equal all-to-all MS operations, interleaved with single-qubit gates that act only on one special qubit. Using a connection with quantum signal processing techniques, we find that it is possible to perform an arbitrary SU(2) rotation on the special qubit if and only if all other qubits are in the state $|1\rangle$. Such controlled rotation gates with $N-1$ control qubits require $2^N$ applications of the MS gate, and can be mapped to a conventional Toffoli gate by demoting a single qubit to ancilla.

1. Introduction

Multiple-control gates arise in various quantum information processing tasks, ranging from basic arithmetic [1] to Grover’s search method [2], which famously requires a gate that performs an inversion around just a single specific state. The prototypical example is the Toffoli$_N$ gate, which performs a bitflip (Pauli-$X$) on the target qubit if and only if $N-1$ control qubits are in the state $|1\rangle$.

Albeit conceptually simple, gates with many control qubits remain challenging to implement in current experimental quantum computers. Because single-qubit gates are typically more accurate than entangling operations, an important challenge is to decompose multiqubit operations into circuits with as few as possible two-qubit gates. To indicate, the Toffoli$_3$ requires at least 5 two-qubit gates [3], or when the CNOT gate is the only available entangling gate, then at least 6 CNOTs are needed [4]. For a larger number of qubits, the required number of basic operations or ancilla qubits grows steeply. The Toffoli$_N$ on $N$ qubits can be constructed through a circuit of depth $O(\log(N))$, requiring $O(N)$ ancilla bits. With just a single ancilla, the best known circuits require $O(N)$ two-qubit gates [5–7]. If no ancillas may be used, then a quadratic number of CNOTs are required [8]. The size of these circuits has been prohibitive in scaling up quantum algorithms on current quantum computer prototypes: even though various systems with a few tens of qubits have been reported, the largest multiple-controlled gate ever performed is, to our best knowledge, the Toffoli$_4$ [9].

A possible workaround is to replace operations by interactions between multiple qubits that natively arise in a quantum computer [10–14]. Quantum computers based on trapped ions typically deal most naturally with the Mølmer–Sørensen gate (MS) as basic entangling gate, which effectively applies an Ising interaction to all pairs of qubits for a specific amount of time [15–18]. For a small number of qubits, fidelities of over 99% have been reported [19–21], and a large body of scientific work focuses on optimizing quantum circuits for this gate [22–27]. To compare to the numbers of the Toffoli$_N$ gate, the best result we are aware of decomposes this operation into $3N-9$ MS operations interleaved with single-qubit
rotations [26], requiring $\frac{N+2}{2}$ ancillary qubits and assuming that each operation may act on a different subset of qubits.

1.1. Overview of results

In this work, we consider a gate set consisting of an all-to-all MS gate that always acts on all $N$ qubits, together with arbitrary single-qubit rotations on just a single special qubit. Our main result is that this formalism allows one to perform rotations on the special qubit that depend on the state of the other qubits. We pay particular attention to the controlled rotation gate, which performs a single-qubit rotation on the special qubit if and only if all other qubits are in the state $|1\rangle$.

More precisely, we define the Mølmer–Sørensen gate acting on all $N$ qubits in the system as

$$\text{MS}(\tau) = \exp \left( -i\frac{\tau}{4} \sum_{j=1}^{N} X_jX_k \right),$$

where $X_j, Y_j, Z_j$ denote the Pauli matrices acting on qubit $j$. In our case, $\tau$ is fixed for a given number of qubits $N$. Moreover, we denote the conventional single-qubit rotations as

$$R_x(\alpha) = \exp \left( -i\frac{\alpha}{2} X \right), \quad R_y(\alpha) = \exp \left( -i\frac{\alpha}{2} Y \right), \quad R_z(\alpha) = \exp \left( -i\frac{\alpha}{2} Z \right).$$

Then, the following circuit implements the operation $R_x(\alpha)$ on a target qubit when all other qubits (the controls) are in the state $|1\rangle$:

$\begin{align*}
&= \text{Repeat for } j = L \text{ down to } 1 \\
&= R_x(\alpha) \quad \text{MS}(\tau) \quad R_x(-\phi_j) \quad R_x(h) \quad R_x(\phi_j) \quad R_x(\phi_0)
\end{align*}$

(3)

Here, the top 3 lines represent $N-1$ control qubits, and a total of $L$ applications of the MS gate occur. $H = \frac{1}{\sqrt{2}}(X + Z)$ represents the Hadamard gate. The parameters $\tau, h$ are always $+\pi/N$ and $-\pi/N$ respectively, and the number of MS pulses needed is, in this case, $L = 2N$. The remaining unknown parameters, $\phi_j$, are discussed below. We call the resultant controlled rotation $C^{N-1}R_x(\alpha)$. Note that using a local basis transformation on the target qubit, this can be turned into any controlled-SU(2) operation. More generally, a similar circuit can implement any rotation $R_x(\alpha_q)$ on the target, where the rotation angle $\alpha_q$ depends on the number of control qubits in the state $|1\rangle$. Such operations cost $L = 4N$ applications of the MS gate.

There is an important difference between $C^{N-1}R_x(\pm \pi)$ (a controlled $\mp iX$ rotation), and the conventional Toffoli$_N$ gate, because the factor of $i$ is not a global phase. In equation (13), we describe how the conventional Toffoli$_{N-1}$ can be retrieved by demoting one of the qubits to ancilla, following [14].

The main body of this paper is devoted to linking the above circuit to the theory of equiangular composite gates, as introduced by Low, Yoder and Chuang [28]. This framework allows the construction of a single-qubit unitary operation $U(\theta)$ that depends in a complicated way on some parameter $\theta$, using a sequence of elementary gates consisting of $R_x(\theta)$ and arbitrary $\theta$-independent gates. The technique to efficiently calculate the appropriate gate sequence is often called signal processing. It proved useful in many recent breakthroughs in the design of quantum algorithms, such as quantum singular value transformations [29, 30], linear combinations of unitaries [31], and efficient Hamiltonian simulation [32, 33].

In this paper, we use signal processing in a very different context. Our core result is that the action of the MS gate can be interpreted as $R_x(\theta_q)$ on a special qubit, where $\theta_q$ depends on the number of ones among the other qubits. With this interpretation we can readily apply a known algorithm to efficiently compute the angles $\{\phi_0, \phi_1, \ldots, \phi_L\}$ that implement the required gate, depending on $N$ and $\alpha$.

The theory and implementation of quantum signal processing is rather involved, and to lower the barrier to apply our results in practice, we provide Python code that calculates these parameters in reference [34]. In particular, the function crot_angles $(N, \alpha)$ returns precisely the list of angles $\phi_j$, which can be readily plugged into the circuit in equation (3). Moreover, in appendix A, we list an explicit circuit for $C^2R_x(\pi)$ and tabulate values of the angles $\Phi$ for $N = 3, 4, 5, 6$. This should allow anyone to use our results without the need to understand all the details of this paper.
2. Composite gate techniques

This section reviews the theory of quantum signal processing and composite gates. We follow the notation from Low, Yoder and Chuang [28], with some adjustments to make it consistent with the algorithm by Haah [35] and our own notation of the $\mathbb{MS}$ gate.

Let $\Phi = \{\phi_0, \phi_1, \ldots, \phi_L\}$ be a list of angles (i.e. real numbers that we interpret to be periodic with period $4\pi$) of length $L + 1$. We can then define a single-qubit rotation of the form

$$F^\Phi(\theta) = R_z(\phi_0) \prod_{j=1}^{L} R_z(-\phi_j)R_x(\theta)R_z(\phi_j).$$  \hspace{1cm} (4)

In other words, the function $F^\Phi$ maps an angle $\theta \in [0, 4\pi)$ to an element of $SU(2)$, a unitary matrix with determinant 1. Each of the $\theta$-dependent steps $R_x(-\phi_j)R_x(\theta)R_x(\phi_j)$ corresponds to a rotation around a vector pointing along the equator of the Bloch sphere, where this vector is given by $\cos(\frac{\phi_j}{2})X - \sin(\frac{\phi_j}{2})Y$.

Any such matrix $F^\Phi(\theta)$ can also be decomposed in a basis of Pauli matrices$^6$,

$$F^\Phi(\theta) = A(\theta)\mathbb{1} + iB(\theta)X + iC(\theta)Y + iD(\theta)Z.$$  \hspace{1cm} (5)

Here, $A, B, C, D$ are real functions of $\theta$. They have to be $4\pi$ periodic in $\theta$, and by unitarity, they satisfy the normalization condition $A(\theta)^2 + B(\theta)^2 + C(\theta)^2 + D(\theta)^2 = 1$. Moreover, it was found that $A$ and $D$ are symmetric functions of $\theta$, whilst $B, C$ are anti-symmetric in $\theta$ (i.e. $A(\theta) = +A(-\theta)$ and $C(\theta) = -C(-\theta)$) [35].

With the above properties, it is convenient to express $A, B, C, D$ in their cosine or sine series. As a technical detail, an expansion over $4\pi$ periodic function takes the form $A = \sum_{k=0}^{L} a_k \cos(k\theta/2)$, but due to the structure of $SU(2)$, only frequencies of $k$ odd (when $L$ is odd) or $k$ even (when $L$ is even) can have nonzero coefficients $a_k$ (and likewise for $B, C, D$).$^7$ We choose to work only with the simpler case of $L$ even, where all functions become $2\pi$ periodic in $\theta$:

$$A(\theta) = \sum_{k=0}^{L/2} a_k \cos(k\theta), \quad B(\theta) = \sum_{k=1}^{L/2} b_k \sin(k\theta)$$
$$C(\theta) = \sum_{k=1}^{L/2} c_k \sin(k\theta), \quad D(\theta) = \sum_{k=0}^{L/2} d_k \cos(k\theta)$$  \hspace{1cm} (6)

In this notation, the degree of the series $L/2$ turns out to be half of the number of $\theta$-dependent steps in equation (4) [28].

The reverse is also true: for any set of series $A, B, C, D$ that is properly normalized and has largest degree $L/2$, there exists a sequence of angles $\Phi$ of length $L + 1$ such that equation (4) holds. Retriving these angles $\Phi$ from known series $A, B, C, D$ can be done efficiently on a classical computer using the algorithm by Haah [35], which is implemented in Python in reference [36]. During the completion of this manuscript, two alternative approaches to retrieve $\Phi$ were announced [37, 38].

This backwards-engineering step is precisely what we will exploit. In section 3 we indicate how $\mathbb{MS}$ pulses can be interpreted as a single-qubit rotation of the form $R_x(\theta_q)$ on the target qubit, where $\theta_q$ depends on the state of the control qubits. Then, in section 4, we find achievable cosine and sine series $A, B, C, D$ that correspond precisely to the gate we aim to obtain, with degrees as low as possible. In fact, we only have to supply the algorithm with $A$ and $B$, as it determines a suitable $C$ and $D$ by itself using the normalization and even/oddness.

The remainder of this section consists of technical notes that one might want to skip on a first read.

Firstly, we note that Haah’s algorithm is slightly more general, allowing $A$ to be chosen either even or odd, and similar (independently) for $B$. The downside is that in such cases, the $\theta$-dependent single-qubit gates $R_x(-\phi_j)R_x(\theta)R_x(\phi_j)$ have to be replaced by $\theta$-rotations around an arbitrary vector on the Bloch sphere.

---

$^5$ Note that the rotations in equation (2) have periodicity $4\pi$ and not $2\pi$. To indicate, on input $\alpha = 2\pi\theta$ each rotation becomes $\frac{\alpha}{\theta}$. This makes a difference because we consider controlled operations.

$^6$ Note that there exist two different conventions for expanding $F^\Phi(\theta)$ into Pauli matrices: we follow the notation by Haah [35], whereas Low et al work with slightly different labeling $ABCD \rightarrow ACDB$ [28, 32, 33]. This is equivalent to a change of basis on the Pauli matrices.

$^7$ To indicate, $R_x(\theta + 2\pi) = -R_x(\theta)$. Therefore, for sequences with an even number of applications of $R_x(\theta)$, the functions $A, B, C, D$ are symmetric under a $2\pi$ shift in $\theta$. An odd number of applications require basis functions to be anti-symmetric under a $2\pi$ shift.
Because we did not find more efficient results in this formalism, we choose to stick with the simpler notation introduced above. Still, future work might exploit these extensions to improve on our results.

We will later see that, to obtain the advertised $C^{N-1}R_z(\alpha)$ gate, it is essential that we have control over the function $D$, which does not directly serve as input. However, we will ensure that $D$ is non-zero only at the special points $\theta = 0$ or $\theta = \pi$, where the anti-symmetric functions $B, C$ are always 0. Exploiting $D^2 = 1 - A^2$ at these points, we can produce $R_z(\alpha)$ by only pinning $A$ to a specific value.

3. Mølmer–Sørensen in the context of single-qubit composite gates

In this section, we argue how the rotation $R_z(\theta)$ (acting on a single qubit, but having $\theta$ as free parameter) can be related to the MS operation (acting on $N$ qubits). We will see that a circuit of the form equation (3) can be broken down into invariant two-dimensional subspaces. The operation on each subspace can be described by $F^\theta(\theta_j)$, where $\theta_j$ depends on the state of the control qubits.

We consider a set of $N$ qubits, which we label by $|N\rangle = \{0, 1, \ldots, N-1\}$. The MS$(\tau)$ operation is interpreted as the evolution of a certain Ising model under Schrödinger’s equation for a time $\tau$. For now, let us analyze the Ising model in the $Z$-basis rather than the $X$-basis, and allow for arbitrary interaction strength $w_{jk}$ between any pair $j, k \in |N\rangle$ of qubits. The corresponding Hamiltonian reads

$$H_{\text{Ising}} = \frac{1}{2} \sum_{j \neq k \in |N\rangle} w_{jk} Z_j Z_k.$$  

The variable $f$ sets an energy scale. Interactions of this form have been experimentally observed between $N = 53$ atoms in a Paul trap [39]. In our case, we refer to qubit 0 as our special target qubit, and the other qubits will be called control qubits. We denote our quantum states in the computational basis as $|b_0, \vec{b}\rangle$, where $b_0 \in \{0, 1\}$ represents the state of the target, and $\vec{b} \in \{0, 1\}^{N-1}$ denotes the states of the control qubits. The states $|b_0, \vec{b}\rangle$ are the eigenstates of $H_{\text{Ising}}$, whose energies we denote by $E_{b_0, \vec{b}}$.

When we include arbitrary rotations on the target qubit, the Hilbert space decomposes into conserved subspaces, each of which can be labeled by the state $\vec{b}$ of the control qubits:

$$\mathcal{H}_b = \text{span}(|0, \vec{b}\rangle, |1, \vec{b}\rangle).$$

Within each of these subspaces, the Ising Hamiltonian $H_{\text{Ising}}$ acts as

$$H_b = \left( \begin{array}{cc} E_{0, \vec{b}} & 0 \\ 0 & E_{1, \vec{b}} \end{array} \right) = \left( \begin{array}{cc} \Delta_{\vec{b}}/2 & 0 \\ 0 & -\Delta_{\vec{b}}/2 \end{array} \right) + E_{\vec{b}} 1,$$

where we defined the energy gap $\Delta_{\vec{b}} = E_{0, \vec{b}} - E_{1, \vec{b}}$ and the mean energy $E_{\vec{b}} = \frac{E_{0, \vec{b}} + E_{1, \vec{b}}}{2}$.

Now, consider the special configuration of couplings

$$w_{0,k} = 1 \quad \forall k \in \{1, \ldots, N-1\} \quad \text{(Star couplings)}$$

$$w_{j,k} = 0 \quad \forall j \geq 1.$$  

This gives rise to a star-shaped connectivity, where all control qubits are coupled to the target, but not among each other. Now, $H_{\text{Ising}}$ has a spectrum as indicated in figure 1: the ground energy is $E = -f(N-1)/2$ when all control qubits are different from the target, and +f energy is added for each control qubit that is the same. The complete eigensystem is

$$\{|0, \vec{b}: |\vec{b}| = q\} \quad E_{0,\vec{b}} = -f \left( \frac{N-1}{2} - q \right) \quad \text{(Star couplings)}$$

$$\{|1, \vec{b}: |\vec{b}| = q\} \quad E_{1,\vec{b}} = f \left( \frac{N-1}{2} - q \right)$$

where $|\vec{b}| = q \in \{0, \ldots, N-1\}$ denotes the Hamming weight (i.e. the number of ones) of the bitstring $\vec{b} \in \{0, 1\}^{N-1}$. For this configuration, $H_b$ takes a particularly simple form, as the gap $\Delta_{\vec{b}}$ is given by

$$\Delta_{\vec{b}} = f(N - 1 - 2q),$$  

This configuration can be used for implementing the $C^{N-1}R_z(\alpha)$ gate as follows: The Hamiltonian $H_b$ is set to act only on the target qubit, and the couplings $w_{jk}$ are chosen such that the ground state is $|0, \vec{b}\rangle$. Then, the state is evolved for a time $\tau$ and projected back onto the target qubit, and this process is repeated for each control qubit in turn. The result is that the state is rotated into the target qubit, and the whole process is repeated until the desired number of rotations has been performed.
We collect the rotation angles the state control qubits. Pinning the rotation energy vanishes for all subspaces, i.e. that can occur due to a combination of MS(For later convenience, after each Here, $H$ hence is the same for all subspaces corresponding to the same Hamming weight $\bar{w}$ regardless of the qubit state. Thus, with change as a response to this. In fact, the only parameter of $H_{\bar{b}}$ that changes is $E_{\bar{b}}$, leading to a subspace-dependent overall shift in energy.

The link between $H_{\text{ising}}$ and the MS gate is as follows. Choosing uniform all-to-all couplings ($w_{jk} = 1$) and unitarily evolving for a time $t = \tau/J$ precisely implements the MS($\tau$) gate, up to a rotation between the $X$ and $Z$ eigenbases:

$$MS(\tau) = H^{\otimes N} \cdot \exp(-iH_{\text{ising}}\tau/J) \cdot H^{\otimes N}$$

Here, $H$ denotes the Hadamard gate. Within the conserved subspace $\mathcal{H}_{\bar{b}}$, up to an overall phase, this operation can be written as

$$MS(\tau) \sim R_x(\Delta_{\bar{b}}\tau/J) e^{-iE_{\bar{b}}\tau/J}$$

For later convenience, after each MS gate we allow a single-qubit rotation $R_x(h)$ to be applied on the target qubit. The combined rotation takes the form

$$MS(\tau)R_x(h) \sim R_x(\theta_q) e^{-iE_{\bar{b}}\tau/J} \quad \text{where} \quad \theta_q = \frac{\Delta_{\bar{b}}\tau}{J} + h = (N-1-2q)\tau + h.$$  

We collect the rotation angles $\theta_q$ using the set notation $\Theta = \{\theta_q\}_{q=0}^{N-1}$, representing all the rotation angles $\theta_q$ that can occur due to a combination of MS($\tau$) and $R_x(h)$. The free parameters $\tau, h$ allow us to spread and shift these relevant angles over the unit circle.

This approach connects the circuit in equation (3) with the composite gate defined in equation (4): the function $F^\tau(\theta)$ can be implemented by a sequence of MS gates (that effectively perform $R_x(\theta)$) interleaved with $R_x(\phi_j)$ rotations. The angles $\theta$ that can occur are precisely those in $\Theta$, which depend on the state of the control qubits. Pinning the rotation $F^\tau(\theta_q)$ that takes place on the target qubit whenever the control qubits have Hamming weight $q$ is equivalent to pinning the values $A(\theta_q), B(\theta_q), C(\theta_q), D(\theta_q)$.

A last detail is the subspace-dependent phase due to $E_{\bar{b}}$, which can be straightforwardly tracked in the case of equal all-to-all interaction ($w_{jk} = 1$ for $j, k$), as is the case in our definition of the MS gate. The permutation symmetry among qubits makes it straightforward to calculate the $N$ unique values of $E_{\bar{b}}$. However, in the context of the circuit in equation (3), we follow a more intuitive approach. We observe that all the terms $X_jX_k$ in the MS gate commute. Moreover, for the control qubits, each subsequent $X_jX_k$ pulse is not interleaved with single-qubit gates, hence rotation angles are additive. Each of these terms associates a phase to a pair of qubits $(j, k)$. When $\tau$ and $L$ are chosen such that $\tau L = 2\pi$, each of these phases reset regardless of the qubit state. Thus, with $\tau = \pi/N$, any choice of $L$ that is a multiple of $2N$ guarantees that subspace-dependent phases $E_{\bar{b}}$ reset in our circuit.

*Because MS($\tau$) and $R_x(h)$ commute, they can be performed in any order, or even simultaneously.*
4. The full composite gate

Let us now consider the functions $A, B, C, D$ that implement a controlled rotation. The gate $C^{N-1}R_z(\alpha)$ corresponds to the choice

$$
\begin{align*}
\text{if } q \neq N - 1: & \quad A(\theta_q) = 1, \\
\text{if } q = N - 1: & \quad A(\theta_q) = \cos(\alpha/2), \quad D(\theta_q) = \sin(\alpha/2).
\end{align*}
$$

(12)

We first choose convenient values values for $A$ and $D$ such that $A(\theta_q) \leq 1$, we restrict the derivatives $A'(\theta_q) = 0$. This way, whenever $A(\theta_q) = 1$, this corresponds to a maximum of the function $A$. An additional advantage is that slight over- or under-rotations of $\theta$ due to an inaccurate MS gate do not affect the resulting operation, to first order in $\theta$. Exploiting again the symmetries of $A$, the only values of $q$ for which these derivatives are relevant are $\lfloor N/2 \rfloor$ up to $q = N - 1$, because the cosine series already has zero derivative at the points $\theta = 0, \pi$. All in all, we have $N$ constraints on the function $A(\theta)$, hence we can parametrize it as a series of degree $N - 1$ (because the series starts counting at $k = 0$).

With this choice of $A(\theta)$, any choice of $B, C, D$ that respects the normalization condition $A^2 + B^2 + C^2 + D^2 = 1$. To enforce $|A(\theta)| \leq 1$, we restrict the derivatives $A'(\theta_q) = 0$. This way, whenever $A(\theta_q) = 1$, this corresponds to a maximum of the function $A$. An additional advantage is that slight over- or under-rotations of $\theta$ due to an inaccurate MS gate do not affect the resulting operation, to first order in $\theta$. Exploiting again the symmetries of $A$, the only values of $q$ for which these derivatives are relevant are $\lfloor N/2 \rfloor$ up to $q = N - 1$, because the cosine series already has zero derivative at the points $\theta = 0, \pi$. All in all, we have $N$ constraints on the function $A(\theta)$, hence we can parametrize it as a series of degree $N - 1$ (because the series starts counting at $k = 0$).

Unfortunately, such fits would generally not respect the normalization condition $A^2 + B^2 + C^2 + D^2 = 1$. To enforce $|A(\theta)| \leq 1$, we restrict the derivatives $A'(\theta_q) = 0$. This way, whenever $A(\theta_q) = 1$, this corresponds to a maximum of the function $A$. An additional advantage is that slight over- or under-rotations of $\theta$ due to an inaccurate MS gate do not affect the resulting operation, to first order in $\theta$. Exploiting again the symmetries of $A$, the only values of $q$ for which these derivatives are relevant are $\lfloor N/2 \rfloor$ up to $q = N - 1$, because the cosine series already has zero derivative at the points $\theta = 0, \pi$. All in all, we have $N$ constraints on the function $A(\theta)$, hence we can parametrize it as a series of degree $N - 1$ (because the series starts counting at $k = 0$).

With this choice of $A(\theta)$, any choice of $B, C, D$ that respects the normalization condition is automatically a valid choice for the $C^{N-1}R_z(\alpha)$ gate. Firstly, at the points $\theta_q$ for $q \neq N - 1$, we have set $A = 1$ and hence at those points, $B = C = D = 0$. Moreover, the point $\theta_{N-1} = \pi$ is a zero for any sine series, so normalization

Note that other choices, such as setting $\theta_{q=N-1} = 0$ would work equally well.

We use $\lfloor a \rfloor$ to denote that $a$ is rounded down to the nearest integer. Similarly, $\lceil a \rceil$ denotes rounding up.
requires \(|D(\theta_{N-1})| = \sin(\alpha/2)|\).\(^{11}\) The behavior of \(B, C, D\) at points other than \(\theta_q\) is irrelevant for our purposes.

### 4.2. From cosine series to composite gate

Having found the series representation of \(A\) and \(B\), we can input these in the algorithm by Haah \([35]\). It finds the angles \(\Phi\) that allow us to form the circuit in equation (3) with \(L = 2N - 2\).

#### 4.3. On relative phases due to \(E_i\)

In its current form, the circuit in equation (3) that uses a sequence of length \(L = 2N - 2\) will also introduce subspace-dependent phases due to \(E_i\). The simplest solution is to round \(L\) up to \(L = 2N\) by adding two addition steps with \(\phi_{2N-1} = 0\) and \(\phi_{2N} = \pi\). These two extra steps have no net effect on the target qubit, but make sure that the \(\text{MS}\) gate performs a full \(2\pi\) rotation on all pairs of control qubits.

#### 4.4. Turning the \(C^{N-1}R_z(\alpha)\) into a conventional \(\text{Toffoli}\)

The rotation of the form \(C^{N-1}R_z(\pi)\) fundamentally differs from a conventional \(\text{Toffoli}_N\) or controlled-\(Z\) operation due to the additional phase \(i\) that appears if and only if the target qubit is rotated. To get rid of this extra factor in the subspace where \(q = N - 1\), we propose the following circuit, as inspired by reference [14]:

\[
\begin{array}{cccccccccc}
\text{Qubit} & & & & & & & & & \\
|0\rangle & \rightarrow & H & \rightarrow & R_z(2\pi) & \rightarrow & |0\rangle & = & & \\
\end{array}
\]

The philosophy here is that the \(C^{N-1}R_z(2\pi)\) gate applies the operation \(\text{Id}\) if all control qubits are \(|1\rangle\) and the operation \(\text{Id}\) otherwise. Hence, this is the same as a controlled-\(Z\) gate acting only on the control qubits. A conjugation by the Hadamard \(H\) on a single qubit maps this into a conventional \(\text{Toffoli}_{N-1}\), with the respective qubit taking the role of target.

### 5. Discussion and possible extensions

The number of Ising pulses \(L = 2N\) needed to implement the \(C^{N-1}R_z(\alpha)\) is very modest and does not depend on \(\alpha\). Still, for small \(N\), there exist highly optimized decompositions that are more optimal than our results. For example, references \([24, 26]\) find decompositions of, respectively, the \(N = 3\) and \(N = 4\) \(\text{Toffoli}\) gates into a mere 3 \(\text{MS}(\tau)\) pulses (albeit with \(\tau\) not equal for all pulses). Moreover, the \(C^2R_z(\pi)\) operation can be implemented in just 4 \(\text{CNOT}\) operations\(^{12}\).

Our results are of main interest thanks to the favourable scaling with increasing \(N\). The best similar result we are aware of is reference \([26]\). It describes circuits for the \(\text{Toffoli}_N\) gate that use \(3N - 9\) \(\text{MS}\) gates and \(N^{2/3}\) ancillae for even \(N\), and \(3N - 6\) \(\text{MS}\) gates and \(N^{-1}N^{-4}\) ancillae when \(N\) is odd, both of which were major improvements over previously known results. Each of these \(\text{MS}\) gates has to act on varying subsets of qubits, and the rotation angle \(\tau\) can differ per step. Using the circuit in equation (13), our proposal implements the \(\text{Toffoli}_N\) using \(2(N + 1)\) \(\text{MS}\) gates, significantly improving the scaling with \(N\) and requiring a mere 1 ancilla. Moreover, our proposal has the added advantage that all \(\text{MS}\) gates are equal: they act equally on all pairs of qubits, and require the same rotation angle \(\tau\) in each gate. This simplicity is an experimental advantage, putting less constraints on control hardware and requiring fewer unique \(\text{MS}\) pulses to be optimized \([41]\).

Our proposal is particularly suitable for the trapped ion quantum information processor described in reference \([41]\). Here, the \(\text{MS}\) interactions are equal between all ions since they are mediated by longitudinal center-of-mass motion in the linear ion crystal. Furthermore, single ion addressing is possible with a focused laser beam allowing the implementation of the single-qubit rotations. In this experimental setup, high fidelity \(\text{MS}\) gates have been implemented with up to 14 ions \([42, 43]\). Our scheme remains viable in

---

\(^{11}\) In our case, where we rely on Haah’s algorithm to choose the precise form of \(D(\theta)\), we may end up with the ‘negative’ branch where \(D(\theta_{N-1}) = -\sin(\alpha/2)\). In general, it is straightforward to map our circuit into its inverse: the circuit in equation (3) should be run backwards, and the \(X\) and \(XX\) operations should be conjugated by setting \(\phi_i \rightarrow \phi_i + \pi\).

\(^{12}\) The circuit can be obtained from the textbook \([40]\), p 182, figure 4.9, when discarding the last six gates on the top two qubits. Thanks to an anonymous referee for pointing this out to us.
future array-based trapped ion computers based on ion shuttling and crystal merging [44]. In such a system, MS gates can remain confined to small sub-crystals.

Alternatively, much longer crystals containing > 50 ions could be considered such as in reference [39]. However, in these systems the interactions are mediated by transverse phonons, resulting in couplings of the form $w_{jk} \sim |j - k|^{-\alpha}$ with $0 \leq \alpha \leq 3$ in theory, but $0.5 \leq \alpha \leq 1.8$ in practice.

In general, a non-uniform Ising model would lead to two types of issues. Firstly, we consider non-uniform couplings $w_{jk}$ only between the control qubits. The resultant operation would look like a normal controlled rotation, perturbed by additional $XX$ interactions among the controls. These phases can in principle be undone by a single Ising evolution that affects only the controls and has connection strengths $-w_{jk}$, i.e. the negative values of the previous operation. However, there is no straightforward method of inverting the sign of the interactions for interactions mediated by transverse phonons [17]. Alternatively, one may keep applying non-uniform operations with weights $w_{jk}$ until all the relative phases reset. The problem then reduces to finding multiples of $w_{jk}$ which are all integer multiples of some base strength. It seems likely that newly developed techniques for finding robust gate operations based on multiple laser frequencies could be successfully applied to find such interaction matrices [45, 46].

A second type of issue arises when the connection strengths $w_{jk}$ to the target qubit are no longer equal, which causes a larger set of possible values $\theta_q$. This results in series $A, B, C, D$ with a larger degree, and hence sequences of more MS pulses.

We conclude that our proposal should be within experimental reach on devices that rely on longitudinal phonon interactions, whereas for transversal interactions further optimizations are needed.

From a theoretical perspective, many further optimizations and extensions of our protocol should be possible. For example, much can be learned from combining the intuition in reference [26] and our results. In particular, making $\tau, h$ different per step, and adding single-qubit rotations on the control qubits, might greatly extend the possibilities of our framework. Moreover, in the large-$N$ limit, one might consider functions $A, B, C, D$ that merely approximately implement a required gate, in exchange for a reduction of the number of steps $L$.

From a computer science perspective, we would be very interested to see lower bounds on the number of MS gates needed for certain operations. Moreover, it seems unclear to us how the gates $\text{Toffoli}_N$ and $CN^{-1}R_x(\alpha)$ compare when used in algorithms in practice, and we hope that future research will find how many of each are required in realistic cases. Some results in this direction can be found in reference [6].

As another possible extension of our protocol, assuming again uniform weights $w_{jk} = 1$, one can make controlled rotation gates that depend in a more general way on the Hamming weight of the controls. That is, one could try to implement different functions $A(\theta), B(\theta), C(\theta), D(\theta)$ that act differently for each $\theta_q \in \Theta$ corresponding to a specific Hamming weight. For example, it should be possible to make a gate that performs $R_x(\alpha_0)$ on the target whenever all controls are 0, and $R_x(\alpha_1)$ whenever the controls have weight $q = 1$, etc, by pinning the values $A(\theta_0) = \cos(\alpha_0)$ and $B(\theta_q) = \sin(\alpha_0)$. More generally, one could try to solve for any weight-dependent operation mixing $X$, $Y$ and $Z$ operators, under the constraint that the functions $A, B, C, D$ are properly normalized. This can be a challenging problem and is very similar to filter design problems in discrete time signal processing [47] and as also suggested by [28], one can take advantage of the wealth of existing techniques from this domain to search appropriate solutions.

Lastly, we note that our results may also be of interest for the engineering of related many-body interactions in trapped ion quantum simulators [48, 49] and may be a starting point for implementing more general many-body interactions decomposed out of native quantum gates [50, 51].

6. Conclusion

We make a connection between the all-to-all Mølmer–Sørensen gate and signal processing techniques. Our main result is that controlled rotations of the form $\text{CN}^{-1}R_x(\alpha)$ can be formed by a circuit consisting of $2N$ MS gates plus single-qubit gates that act only on the target qubit. This operation is easily mapped to a $\text{Toffoli}_{N-1}$ gate. Each of the required building blocks has been performed at high fidelity in recent experiments, indicating that our formalism can be realistically applied on near-term quantum computers. We also identify various extensions that may lead to further improvements and generalizations of our protocol.

Acknowledgments

K Groenland and K Schoutens are supported by the QM & QI grant of the University of Amsterdam, supporting QuSoft. R Gerritsma is supported by the Netherlands Organization for Scientific Research (Grant 680.91.120).
Figure A1 shows an explicit circuit for the $C^2R_z(−\pi)$ operation derived using the method presented in this paper. Note that adjacent $R_z(\phi)$ operations have been merged, leading to a more compact form than equation (3). Generalization to larger a number of qubits $N$ is straightforward using the numbers in table A1.

Table A1. The rotation angles $\phi_j$ that implement a controlled-$iZ$ operation with $N = 1$ controls, in the way indicated figure A1. Note that we chose the angle numbered $2N$ to be zero, such that it can be omitted.

<table>
<thead>
<tr>
<th>$\tau = -\hbar$</th>
<th>$\phi_0$</th>
<th>$\phi_1$</th>
<th>$\phi_2$</th>
<th>$\phi_3$</th>
<th>$\phi_4$</th>
<th>$\phi_5$</th>
<th>$\phi_6$</th>
<th>$\phi_7$</th>
<th>$\phi_8$</th>
<th>$\phi_9$</th>
<th>$\phi_{10}$</th>
<th>$\phi_{11}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$N = 3$</td>
<td>$\pi/3$</td>
<td>$-1.855$</td>
<td>$-2.118$</td>
<td>$-0.525$</td>
<td>$-2.118$</td>
<td>$-1.855$</td>
<td>$-\pi$</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$N = 4$</td>
<td>$\pi/4$</td>
<td>$-2.366$</td>
<td>$-1.564$</td>
<td>$1.577$</td>
<td>$1.55$</td>
<td>$-1.564$</td>
<td>$-2.366$</td>
<td>$-\pi$</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$N = 5$</td>
<td>$\pi/5$</td>
<td>$-2.61$</td>
<td>$-1.098$</td>
<td>$1.417$</td>
<td>$-1.116$</td>
<td>$-2.041$</td>
<td>$-1.116$</td>
<td>$1.417$</td>
<td>$-1.098$</td>
<td>$-2.61$</td>
<td>$-\pi$</td>
<td>0</td>
</tr>
<tr>
<td>$N = 6$</td>
<td>$\pi/6$</td>
<td>$-2.745$</td>
<td>$-0.79$</td>
<td>$1.146$</td>
<td>$-1.155$</td>
<td>$0.81$</td>
<td>$2.312$</td>
<td>$0.81$</td>
<td>$-1.155$</td>
<td>$1.146$</td>
<td>$-0.79$</td>
<td>$-2.745$</td>
</tr>
</tbody>
</table>

Appendix A. An example circuit

Figure A1 shows an explicit circuit for the operation $C^2R_z(−\pi)$, i.e. the double-controlled $iZ$ gate. Here, we combined consecutive rotations of the form $R_z(\phi_j)R_z(−\phi_{j-1})$ into a new rotation $R_z(\phi_j)$ to shorten the circuit. For other system sizes $N = 3, 4, 5, 6$, the related angles $\phi_0, \ldots, \phi_N$ for the $C^N-1R_z(−\pi)$ operation can be found in table A1, where again the tilde indicates that adjacent rotations with angles $\phi_j = \phi_{j-1}$ have been combined.

ORCID iDs

Koen Groenland © https://orcid.org/0000-0002-9474-2130
Freek Witteveen © https://orcid.org/0000-0002-3731-4455
Kareljan Schoutens © https://orcid.org/0000-0002-4159-877X
Rene Gerritsma © https://orcid.org/0000-0002-3466-5719

References


Maslov D 2017 Basic circuit compilation techniques for an ion trap quantum machine New J. Phys. 19 023035


[33] Groenland K 2019 Github: Controlled Rotation Quantum Gates Using Trapped Ions

[34] Haah J 2019 Product decomposition of periodic functions in quantum signal processing Quantum 3 190


