# Parallelism for quantum computation with qudits

Dianne P. O'Leary,[1,3,*] Gavin K. Brennen,[2,†] and Stephen S. Bullock[3,‡]
[1]*Department of Computer Science and Institute for Advanced Computer Studies, University of Maryland, College Park, Maryland 20742, USA*
*and Mathematical and Computational Sciences Division, National Institute of Standards and Technology, Gaithersburg, Maryland 20899, USA*
[2]*Institute for Quantum Optics and Quantum Information of the Austrian Academy of Sciences, A-6020, Innsbruck, Austria*
[3]*IDA Center for Computing Sciences, 17100 Science Drive, Bowie, Maryland 20715-4300, USA*
(Received 24 March 2006; published 28 September 2006)

Robust quantum computation with $d$-level quantum systems (qudits) poses two requirements: fast, parallel quantum gates and high-fidelity two-qudit gates. We first describe how to implement parallel single-qudit operations. It is by now well known that any single-qudit unitary can be decomposed into a sequence of Givens rotations on two-dimensional subspaces of the qudit state space. Using a coupling graph to represent physically allowed couplings between pairs of qudit states, we then show that the logical depth (time) of the parallel gate sequence is equal to the height of an associated tree. The implementation of a given unitary can then optimize the tradeoff between gate time and resources used. These ideas are illustrated for qudits encoded in the ground hyperfine states of the alkali-metal atoms $^{87}$Rb and $^{133}$Cs. Second, we provide a protocol for implementing parallelized nonlocal two-qudit gates using the assistance of entangled qubit pairs. Using known protocols for qubit entanglement purification, this offers the possibility of high-fidelity two-qudit gates.

DOI: 10.1103/PhysRevA.74.032334 PACS number(s): 03.67.Lx

## I. INTRODUCTION

Quantum computation requires the ability to process quantum data on a time scale that is small compared to the error rate induced by environmental interactions (decoherence). Robust computation results when the rate of error in the control operations and the rate of decoherence are below some threshold independent of the size of the computational register. The threshold theorem implies such rates exist, but it assumes arbitrary connectivity between subsystems as well as the ability to implement the control operations with a high degree of parallelism [1], allowing additional resources (e.g., lasers) to be used to reduce time. Quantum-computer architectures, therefore, should be designed to support parallel gate operations and measurements. At the software level some work has been done regarding parallel computation with qubits. For example, certain quantum algorithms such as the quantum Fourier transform can be parallelized [2], and there are techniques to compress the logical depth (i.e., time) of a quantum circuit on qubits using the commutativity of gates in the Clifford group [3]. Further, by using distributed entanglement resources, some frequently used control operations can be parallelized [4].

This work concerns parallel unitary operations on qudits, i.e., $d$-level systems where typically $d>2$. There are several reasons for considering such systems. Many physical candidates for quantum computation with qubits work by encoding in a subspace of a system with many more accessible levels. Control over all the levels is important for state preparation, simulating quantum processes, and measurement. In particular, encoding in decoherence-free subspaces usually involves control over multiple distinguishable states. Additionally, for small quantum computations, a fixed unitary $U \in U(d)$ for $d$ small but larger than 2, can often be implemented with higher fidelity in a single qudit rather than by simulation with two-qubit gates. Further, at the level of tensor structures, some quantum processing may be more efficient with qudits, e.g., the Fourier transform over an Abelian group whose order is not divisible by 2 [5]. It is straightforward to show that naive qubit emulation of qudits is inefficient [6].

Fast single-qudit gate times are important in order to implement quantum error correction before errors accumulate [7]. In Sec. II we derive *parallel* implementations of general one-qudit unitary gates, where the quantum one-qudit gate library is restricted to a small set of couplings between two-dimensional subspaces (Givens rotations). The choice of this Givens library of one-qudit gates reflects standard coupling diagrams, i.e., the particular rotations obey selection rules in the physical system that encodes the qudit. Prior work considered minimum-gate circuits for such generalized coupling diagrams but did not further optimize these circuits in terms of depth [8]. Parallelism is possible because quantum gates acting on disjoint subspaces (i.e., qudits in different states) can be applied simultaneously, at the expense of additional control resources. Our method is particularly helpful for experimental implementations because it can be applied to a large class of systems with different allowed physical couplings. We provide examples for qudit control with ground electronic hyperfine levels of $^{87}$Rb and $^{133}$Cs and show that it is possible to achieve impressive speedup with these systems using three pairs of control fields.

We hope this work is of interest to experimentalists and others who might not want to study the mathematical formalism on first reading. The heart of Sec. II is two state-

*Electronic address: oleary@cs.umd.edu
†Electronic address: gavin.brennen@uibk.ac.at
‡Electronic address: ssbullo@super.org

 

synthesis examples, explained in Figs. 1–7 and Tables I and II; then Sec. II D shows how to apply unitary transformations using this state synthesis.

Further, in Sec. III we obtain depth-optimized (parallel) implementations of nonlocal two-qudit gates. These operations generically require $O(d^3)$ controlled unitary gates [8]. We show how this can be parallelized to depth $O(d^2)$ using $O(d^3)$ maximally entangled qubit pairs (e-bits). While the protocol is not optimized in terms of e-bits consumed, it is a step forward to the goal of high-fidelity two-qudit gates. This is because the qubit resources can be chosen to be ancillary degrees of freedom of the particle encoding the qudit. Thus they can be prepared in entangled pairs nondeterministically and purified before the nonlocal gate is implemented.

A third aspect of parallelism [2] involves reducing the logical depth of a circuit by judicious grouping of single- and two-particle gates that can be performed at the same time step, assuming connectivity of the particles. This is roughly analogous to classic circuit layouts and will not be considered here.

## II. PARALLELISM IN STATE SYNTHESIS AND UNITARY TRANSFORMATION FOR A SINGLE QUDIT

In typical physical systems encoding a single qudit, arbitrary couplings are not allowed. Whereas we can represent any unitary $U \in U(d)$ as an operator generated from an appropriate set of Hamiltonians, viz., $U=\exp(-i\Sigma_{j=0}^{d^2} t_j h_j)$ where $t_j \in \mathbb{R}$ and $\sqrt{-1}h_j \in \mathfrak{u}(d)$ with $h_j=h_j^\dagger$, it is generally not possible to turn on all the couplings $h_j$ at the same time. It is a problem of quantum control to determine how to simulate a single-qudit unitary using a sequence of available couplings.

Because quantum computations need only be simulated up to a global phase, we restrict ourselves to implementations of a generic unitary $U \in \mathrm{SU}(d)$. One way to implement $U$ is by a covering with gates generated by the $\mathfrak{su}(2)$ subalgebras $\mathfrak{g}_{j,k}$ acting on the subspaces spanned by the state pairs $(|k\rangle, |j\rangle)$:

$$\mathfrak{g}_{j,k} = \{i\lambda_{j,k}^{x,y,z};\ \lambda_{j,k}^{x} = |j\rangle\langle k| + |k\rangle\langle j|,\ \lambda_{j,k}^{y} = -i(|j\rangle\langle k| - |k\rangle\langle j|),$$
$$\lambda_{j,k}^{z} = |j\rangle\langle j| - |k\rangle\langle k|\}. \quad (1)$$

This is realized by a *QR* decomposition of the *inverse* unitary into a product of unitary (Givens) rotation matrices that reduce it to diagonal form $D^\dagger$:

$$D^\dagger = \left(\prod_{\ell=1}^{d(d-1)/2} G_{j_\ell k_\ell}\right) U^\dagger. \quad (2)$$

Here, each Givens rotation can be chosen to be a function of two real parameters only:

$$G_{jk}(\gamma,\phi) = e^{-i\gamma(\cos\phi\lambda_{j,k}^{x} - \sin\phi\lambda_{j,k}^{y})}. \quad (3)$$

Typically, parameters are chosen so that consecutive Givens rotations introduce an additional zero below the diagonal of the unitary. Thus a sequence of such rotations realizes the inverse unitary up to relative phases, and the reversed sequence of inverse rotations realizes the unitary itself (up to a diagonal gate). There are $d(d-1)/2$ elements below the diagonal; hence the gate count in Eq. (2). The entire synthesis then follows by $U=D(\Pi_{\ell=1}^{d(d-1)/2} G_{j_\ell k_\ell})$. Using a Euler decomposition of SU(2),

$$e^{i\lambda_{j,k}^{z}\beta} = G_{j,k}(-\pi/4,\pi/2)G_{j,k}(\beta,0)G_{j,k}(\pi/4,\pi/2),$$

the diagonal gate can be built using $3(d-1)$ Givens rotations.

A second way to synthesize a unitary transformation is to use a spectral decomposition

$$U = \prod_{\ell=0}^{d-1} W_\ell C_\ell W_\ell^\dagger \quad (4)$$

where $W_\ell$ is a unitary matrix that maps the basis state $|\ell\rangle$ to the eigenvector corresponding to the $\ell$th eigenvalue of $U$, and $C_\ell$ is the identity matrix with its $(\ell,\ell)$ element replaced by the $\ell$th eigenvalue. Each matrix $W_\ell^\dagger$ implements a state-synthesis operation and can be implemented as a product of $G_{jk}(\gamma,\phi)$. In this section we consider parallelism, both in state synthesis and in the two unitary constructions above.

Particular physical systems exhibit symmetries that constrain and refine the broad picture of unitary evolution presented so far [9,10]. This work focuses on systems in which a limited number of pairs of states can be coupled at any given time. The exemplar system is a qudit encoded in the ground hyperfine state of a neutral alkali-metal atom, where the number of pairs that may be coupled at once is determined by the number of lasers incident on the atoms. Other candidate systems for quantum computation, such as flux-based Josephson-junction qudits and electronic states of trapped ions, may allow this type of control.

We recall how the selection rules on an atom with hyperfine electron structure constrain the allowed Givens evolutions of the system [8,11]. A pair of Raman pulses can couple states $|F_\downarrow, M_F\rangle \leftrightarrow |F_\uparrow, M_F'\rangle$. In the linear Zeeman regime, a specific pair of hyperfine states can be addressed by choosing the appropriate frequency and polarization of the two Raman beams. The coupling acts on the electron degree of freedom which imposes a selection rule $\Delta M_F = M_F - M_F' = \pm 2, \pm 1, 0$. To demonstrate the power of our unitary synthesis technique, we restrict discussion to the selection rule $\Delta M_F = M_F - M_F' = \pm 1, 0$. This restriction is valid when the detuning of each Raman laser beam from the excited state is much larger than the hyperfine splitting in the excited state ($\Delta \gg E_{ehf}$) [12]. There is a practical advantage to restricting discussion to this selection rule. Spontaneous emission during the Raman gate scales as $\gamma \sim \Gamma|\Omega_1\Omega_2|/\Delta^2$, where $\Omega_{1,2}$ are the Rabi frequencies of the two Raman beams and $\Gamma$ is the spontaneous emission rate from the excited state. Working in the limit of large detunings reduces errors due to spontaneous scattering events.

The hyperfine levels for a $d=8$ qudit and the induced coupling graph are shown in Figs. 1 and 2. We assume that the amplitude and phase of the Raman beams can be controlled so that each Givens rotation $G_{jk}(\gamma,\phi)$ can be generated in a single time step. Relative phases between a pair of basis states can be achieved by a Euler decomposition of $e^{-i\beta\lambda_{j,k}^{z}}$ using three Givens rotations. Alternately, relative

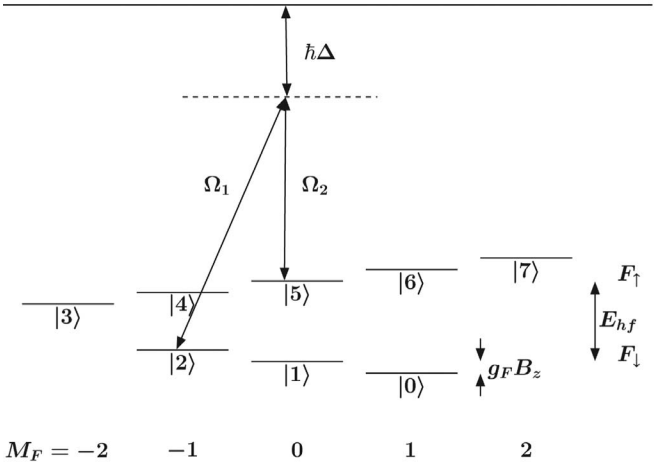


FIG. 1. A single $d=8$ qudit encoded in the ground-state hyperfine levels of $^{87}$Rb. A pair of lasers can couple states in different hyperfine manifolds according to the selection rule $\Delta M_F=0,\pm 1$.

phases might be introduced using off-resonant Raman beams. Basis states can be phased using ac Stark shifts induced by off-resonant $\pi$-polarized light, or time-dependent magnetic fields. Whatever the technique used it is crucial to keep track of phases accumulated on all the basis states. For unitary synthesis using Given's rotations, phases on states other than the addressed pair can be corrected for in subsequent steps without undoing previous effort (see [8]).

It is notable that, while the multitude of hyperfine levels in atomic systems provides a large state space of quantum-information processing, these states are sensitive to errors. For instance, it is possible to choose disjoint two-dimensional subspaces, spanned by $\{|F_\downarrow,M_F\rangle,|F_\uparrow,-M_F\rangle\}$, that are insensitive to small magnetic field fluctuations along the quantization axis. Fluctuating fields along different axes have negligible effect provided a large enough fixed Zeeman field is applied. There are no such error-avoidance codes when using the entire hyperfine manifold. Hence parallelism, on a scale that can support error correction on a time scale fast compared to environmental noise, will be crucial.

### A. Achieving parallelism in state synthesis

To implement the unitary state-synthesis operator $W_\ell^\dagger$, we construct a sequence of rotations taking a particular vector to a given state $|\ell\rangle$. Again, this is technically the reverse of state synthesis: $W|\ell\rangle=|\psi\rangle$ for a generic pure state $|\psi\rangle$ inverts to a sequence of unitaries $G_{jk}(\gamma,\phi)$ accomplishing $W^\dagger|\psi\rangle=|\ell\rangle$. Thus in the application $|\ell\rangle$ will be the fiducial state, and we attempt to treat all possibilities. We abbreviate the rotation of Eq. (3) by $G_{jk}$.

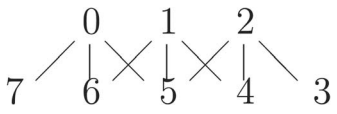


FIG. 2. Coupling graph for $^{87}$Rb. There is an edge between two states if a pair of lasers can couple them as in Fig. 1.

One tool for identifying sequences of rotations that produce $W_\ell^\dagger$ is the rotation or coupling graph, in which node $j$ is connected to node $k$ if a rotation between rows $j$ and $k$ is physically realizable [13], or, in other words, if transitions between the two states $|j\rangle$ and $|k\rangle$ are permitted. Then $W_\ell^\dagger$ is constructed by the sequence of rotations determined by constructing a spanning tree rooted at $\ell$ and successively eliminating leaf nodes by a rotation with their parent. We construct a spanning tree from a graph that has $d$ nodes by keeping exactly $d-1$ edges while maintaining connectivity among all of the nodes. One node $\ell$ is designated to be the root and is put at the top of the diagram. Every node except the root is the child of its parent, the node closer to the root with which it shares an edge. The leaves of the tree are the nodes that have no children.

Consider, for example, the coupling graph of Fig. 2. To perform state synthesis for $|0\rangle$, we can form a spanning tree by breaking the edge between 1 and 5, breaking one of the edges in the cycle 0,5,2,4,1,6,0, and choosing the root to be $|0\rangle$. If we break the edge between 2 and 4, then the resulting tree has three leaves, 7 (eliminated by $G_{07}$), 3 (eliminated by $G_{23}$), and 4 (eliminated by $G_{14}$), We can then eliminate the two resulting leaves 1 and 2, and then 6 and 5. Therefore, we have constructed a rotation sequence

$$G_{05}G_{06}G_{61}G_{52}G_{14}G_{23}G_{07}$$

that synthesizes $|0\rangle$ in seven time steps, hereafter just called *steps*.

To understand the potential for parallelism, note that some of these rotations commute and can therefore be applied in parallel. This is a special case of the assertion that infinitesimal unitaries $ih_1,ih_2\in \mathrm{u}(d)$ may be applied in parallel if and only if $[h_1,h_2]=0$ if and only if $e^{ith_1}$ and $e^{ith_2}$ commute for all $t$ real. We rely on the following result.

*Proposition 1.* A subsequence of $p$ rotations can be applied in parallel if and only if all $2p$ indices are distinct.

*Proof.* It is easy to verify that if all four indices are distinct, then $G_{jk}G_{nm}=G_{nm}G_{jk}$. Conversely, if the four indices are not distinct, then the order of application matters and therefore the rotations cannot be applied in parallel. The result follows by induction on $p$. ■

Using square brackets to group rotations that can be applied in parallel, the seven-step rotation sequence of our example becomes the four-step parallel rotation sequence

$$G_{05}G_{06}[G_{61}G_{52}][G_{14}G_{23}G_{07}]. \tag{5}$$

The next interesting question is how we might determine an ordering of rotations to produce a parallel rotation sequence with a small number of steps. To answer this question, we build upon an algorithm of He and Yesha [[14], Sec. III A]. Given a spanning tree, they create a *binary computation tree* (BCT) by working from the bottom up and replacing every internal node in the spanning tree by a leaf connected to a chain of $p$ nodes, where $p$ is the number of children of the node. They then attach one child to each of the new nodes. The final result is a binary tree. (This process is illustrated in Fig. 3 for a spanning tree of the coupling graph in Fig. 2 rooted at node 3.) The following proposition shows that the number of steps in our parallel rotation se-

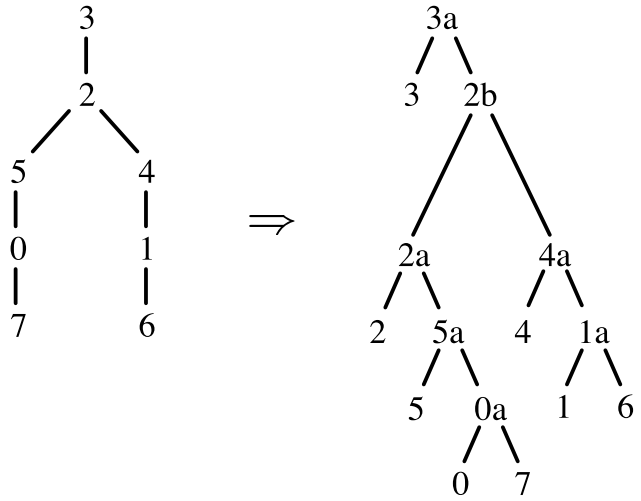


FIG. 3. A spanning tree (left) and a BCT (right) for node 3 of $^{87}$Rb. On the left, the graph in Fig. 2 is redrawn with state 3 at the top, omitting the edges (0,6) and (5,1) in order to remove cycles and make the graph a tree with two leaves, corresponding to $|7\rangle$ and $|6\rangle$. We obtain the BCT on the right by replacing each nonleaf by a leaf connected to a chain of $p$ nodes, where $p$ is the number of children. For example, node 2 in the spanning tree has two children, so at right it is replaced with nodes 2, 2$a$, and 2$b$. We then attach one child to each of the new nodes. The result is that, working from the bottom up, rotations on the same level [for example, (2,5) and (4,1)] can be done simultaneously, accomplishing synthesis of $|3\rangle$.

quence is equal to the height of the BCT, not the height of the spanning tree.

*Proposition 2.* An ordering of the rotations can be obtained by constructing the BCT for a spanning tree of the coupling graph and scheduling each rotation at time step $k-j$, where $k$ is the height of the BCT and $j$ is the distance of the two leaves of the rotation from the root of the BCT. The resulting number of steps is $k-1$.

*Proof.* In constructing the BCT, we have split each node of the spanning tree that is involved in more than one rotation into a chain of nodes, each on a distinct level. This assures that rotations on the same level commute and therefore can be applied in parallel. ■

The resulting ordering is within a factor of $O(\log_2 m)$ of optimal, where $m$ is the number of rotations [14]. We next present a direct (in fact greedy) algorithm which also orders the rotations for optimal parallelism.

At each step, consider each leaf of the spanning tree in order of its distance from the root (more distant leaves first), and process (remove) any leaf whose rotation can be applied

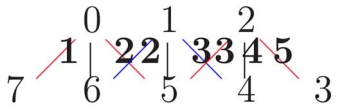


FIG. 4. (Color online) State synthesis timing diagram for $|3\rangle$ for the rubidium alkali-metal atom using two-way parallelism. The time steps, indicated in bold, are taken from the BCT in Fig. 3, working from the bottom up. All transitions are directed toward $|3\rangle$.

TABLE I. Parallel rotation sequences for state synthesis using laser Raman-coupled connections between hyperfine states of $^{87}$Rb. They require five steps for $|3\rangle$ and $|7\rangle$ and four steps for the other kets, rather than the seven steps of the sequential algorithm. The sequence for $|3\rangle$ corresponds to the construction in Fig. 3.

| | |
|---|---|
| $\|0\rangle$ | $G_{05}\,[G_{06}G_{52}]\,[G_{61}G_{07}]\,[G_{14}G_{23}]$ |
| $\|1\rangle$ | $G_{16}\,[G_{60}G_{14}]\,[G_{05}G_{42}]\,[G_{23}G_{07}]$ |
| $\|2\rangle$ | $G_{25}\,[G_{24}G_{50}]\,[G_{41}G_{23}]\,[G_{16}G_{07}]$ |
| $\|3\rangle$ | $G_{32}\,G_{24}[G_{25}G_{41}][G_{50}G_{16}]\,G_{07}$ |
| $\|4\rangle$ | $G_{41}\,[G_{16}G_{42}]\,[G_{25}G_{60}]\,[G_{23}G_{07}]$ |
| $\|5\rangle$ | $G_{50}\,[G_{52}G_{07}]\,[G_{06}G_{24}]\,[G_{61}G_{23}]$ |
| $\|6\rangle$ | $G_{61}\,[G_{14}G_{60}]\,[G_{05}G_{42}]\,[G_{23}G_{07}]$ |
| $\|7\rangle$ | $G_{70}G_{06}\,[G_{61}G_{05}]\,[G_{14}G_{52}]\,G_{23}$ |

in parallel with those already chosen for processing. The two algorithms give the same number of steps but perhaps assign a different timing to some rotations. For example, the greedy algorithm applied to the spanning tree on the left of Fig. 3 yields

$$G_{32}G_{24}G_{25}[G_{50}G_{41}][G_{07}G_{16}],$$

while the BCT on the right of the figure yields the schedule shown in Fig. 4:

$$G_{32}G_{24}[G_{25}G_{41}][G_{50}G_{16}]G_{07}.$$

Both rotation sequences require five steps.

Therefore, we can determine an ordering for the rotations to perform state synthesis for $|\ell\rangle$ by considering in turn each possible spanning tree rooted at $|\ell\rangle$, constructing an ordering for it, and choosing the ordering that provides the smallest number of steps.

It is possible that resource constraints prevent us from implementing a parallel ordering. Suppose for example a limited number of laser beams allows us to apply only two rotations a time. State synthesis for $|0\rangle$ [Eq. (5)] can still be accomplished using a four-step rotation sequence, but it requires a nontrivial rearrangement:

$$G_{05}[G_{06}G_{52}][G_{61}G_{07}][G_{14}G_{23}]. \tag{6}$$

In general, such a constrained scheduling problem is difficult to solve exactly, although good heuristics exist.

### B. Examples of parallelism in state synthesis

We apply our state synthesis algorithms to rubidium and cesium.

*1. Hyperfine levels of $^{87}$Rb.* Only the nine transitions corresponding to the edges of the coupling graph of Fig. 2 are allowed, and the edge between 1 and 5 will not be used in

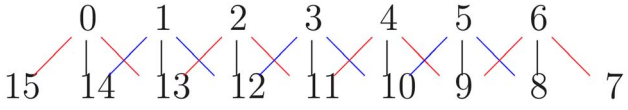


FIG. 5. (Color online) Coupling graph for $^{133}$Cs.

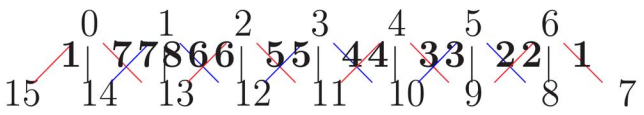


FIG. 6. (Color online) State-synthesis timing diagram for $|1\rangle$ for the cesium alkali-metal atom using two-way parallelism. All transitions are directed toward $|1\rangle$.

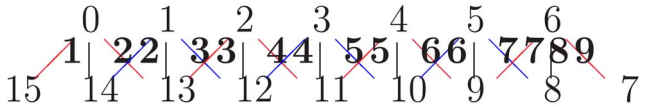


FIG. 7. (Color online) State-synthesis timing diagram for $|7\rangle$ for the cesium alkali-metal atom using two-way parallelism.

our algorithms, since it does not lead to speedup [15].

Optimal parallel rotation sequences, constructed using Proposition 2, are given in Table I.

*2. Hyperfine levels of $^{133}Cs$.* The coupling graph of allowed transitions for $^{133}$Cs is given in Fig. 5. We partition these transitions into three groups.

(i) The *outer chain* of (red) transitions between $|15\rangle$, $|0\rangle$, $|13\rangle$, $|2\rangle$, $|11\rangle$, $|4\rangle$, $|9\rangle$, $|6\rangle$, and $|7\rangle$.

(ii) The *inner chain* of (blue) transitions between $|14\rangle$, $|1\rangle$, $|12\rangle$, $|3\rangle$, $|10\rangle$, $|5\rangle$, and $|8\rangle$.

(iii) A *ladder* of transitions between the two chains.

Since $d=16$, state synthesis requires 15 rotations. If the desired state is $|3\rangle$, for example, then we can use the outer chain of transitions to depopulate $|7\rangle$, $|6\rangle$, $|9\rangle$, $|4\rangle$ (in order) and then $|15\rangle$, $|0\rangle$, $|13\rangle$, $|2\rangle$, and then use the ladder transition from $|11\rangle$ to $|3\rangle$. Similarly, the inner chain of transitions can be used to empty $|14\rangle$, $|1\rangle$, $|12\rangle$, $|8\rangle$, $|5\rangle$, and finally $|10\rangle$. This pattern of using the outer chain, the inner chain, and a single ladder transition accomplishes state synthesis for an arbitrary state.

Complete parallelism is possible in the application of rotations from the outer chain with those in the inner, since no state is involved in both chains. If two rotations can be applied at once, then we need nine steps for state synthesis to $|15\rangle$ or $|7\rangle$ and eight steps for the other kets. We illustrate such a scheme in Figs. 6 and 7, marking each transition with the step at which it is used.

### C. Parallelism in one-qudit unitary transformations

Recall that a state-synthesis algorithm yields algorithms for realizing arbitrary one-qudit unitary evolutions in (at least) two different ways: by invoking the *QR* matrix decomposition [Eq. (2)] or by the spectral theorem [Eq. (4)]. The number of parallel steps for a generic unitary can be significantly greater when using the spectral theorem. For example, for $^{87}$Rb, the spectral decomposition would take 68 steps plus the steps needed to apply the phases. The number of steps to apply parallel *QR* is much less; with three-way parallelism it is at most $2d-3=13$ ($d=8$) plus the steps to apply the phases. Also note that the sequential *QR* requires $d(d-1)/2=28$ steps, so this is a considerable speedup.

A rotation sequence that achieves this bound of 13 steps for *QR* can be constructed using the precedence graph for the computation [16]. Suppose we order the rows as 7,5,0,6,1,4,2,3. We usually use rotations that eliminate an element in any row by a rotation with the element directly above it, but in the first column we use the rotation sequence

$$G_{70}G_{05}G_{06}[G_{52}G_{61}][G_{14}G_{23}].$$

This sequence specifies predecessors for each rotation in the first column. Define the predecessors of a rotation for columns after the first to be the rotations zeroing elements to the south, west, and northwest, if those rotations exist. Each rotation can be performed after all of its predecessors are completed. Therefore, the numerical value of each entry below the diagonal in the following matrix denotes the step at which the entry can be zeroed:

$$\begin{array}{c} 7 \\ 5 \\ 0 \\ 6 \\ 1 \\ 4 \\ 2 \\ 3 \end{array} \begin{bmatrix} x & x & x & x & x & x & x & x \\ 4 & x & x & x & x & x & x & x \\ 5 & 8 & x & x & x & x & x & x \\ 3 & 7 & 9 & x & x & x & x & x \\ 2 & 6 & 8 & 10 & x & x & x & x \\ 1 & 5 & 7 & 9 & 11 & x & x & x \\ 2 & 4 & 6 & 8 & 10 & 12 & x & x \\ 1 & 3 & 5 & 7 & 9 & 11 & 13 & x \end{bmatrix}.$$

Thus, using three-way parallelism, an arbitrary unitary can be applied in 13 steps, plus the steps for phasing.

If only two-way parallelism is allowed, then more steps are necessary. We schedule rotations by cycling through the columns in round-robin order (right to left), scheduling at most one rotation per column, until all rotations are scheduled. If the predecessors of the column's next rotation are scheduled, then that rotation is scheduled for the earliest available time step after their scheduled steps. The resulting time steps are

$$\begin{array}{c} 7 \\ 5 \\ 0 \\ 6 \\ 1 \\ 4 \\ 2 \\ 3 \end{array} \begin{bmatrix} x & x & x & x & x & x & x & x \\ 4 & x & x & x & x & x & x & x \\ 6 & 10 & x & x & x & x & x & x \\ 3 & 8 & 11 & x & x & x & x & x \\ 2 & 7 & 9 & 12 & x & x & x & x \\ 1 & 5 & 8 & 11 & 13 & x & x & x \\ 2 & 4 & 6 & 9 & 12 & 14 & x & x \\ 1 & 3 & 5 & 7 & 10 & 13 & 15 & x \end{bmatrix}.$$

These 15 steps are optimal for two-way parallelism; the last two rotations must be applied sequentially, so the 28 rotations cannot be applied in 14 steps.

A similar construction using the cesium coupling graph shows that at most 29 steps are required using seven-way parallelism. We order the rows as 15,14,0,13,1,12,2,11,3,10,4,9,5,8,6,7. The rotations used in the first column are

$$G_{15,0}G_{0,13}G_{13,2}G_{2,11}G_{11,4}G_{4,9}G_{9,5}G_{9,6} \quad [G_{0,14}G_{13,1}G_{2,12}G_{11,3}G_{4,10}G_{5,8}G_{6,7}],$$

while in other columns we use rotations that eliminate an element in any row by a rotation with the element directly above it. The time steps are as follows:

| | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 15 | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ |
| 14 | 1 | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ |
| 0 | 9 | 16 | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ |
| 13 | 8 | 15 | 17 | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ |
| 1 | 1 | 14 | 16 | 18 | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ |
| 12 | 1 | 13 | 15 | 17 | 19 | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ |
| 2 | 7 | 12 | 14 | 16 | 18 | 20 | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ |
| 11 | 6 | 11 | 13 | 15 | 17 | 19 | 21 | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ |
| 3 | 1 | 10 | 12 | 14 | 16 | 18 | 20 | 22 | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ |
| 10 | 1 | 9 | 11 | 13 | 15 | 17 | 19 | 21 | 23 | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ |
| 4 | 5 | 8 | 10 | 12 | 14 | 16 | 18 | 20 | 22 | 24 | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ |
| 9 | 4 | 7 | 9 | 11 | 13 | 15 | 17 | 19 | 21 | 23 | 25 | $x$ | $x$ | $x$ | $x$ | $x$ |
| 5 | 3 | 6 | 8 | 10 | 12 | 14 | 16 | 18 | 20 | 22 | 24 | 26 | $x$ | $x$ | $x$ | $x$ |
| 8 | 1 | 5 | 7 | 9 | 11 | 13 | 15 | 17 | 19 | 21 | 23 | 25 | 27 | $x$ | $x$ | $x$ |
| 6 | 2 | 4 | 6 | 8 | 10 | 12 | 14 | 16 | 18 | 20 | 22 | 24 | 26 | 28 | $x$ | $x$ |
| 7 | 1 | 3 | 5 | 7 | 9 | 11 | 13 | 15 | 17 | 19 | 21 | 23 | 25 | 27 | 29 | $x$ |

If fewer parallel resources are available, we can again reschedule our steps as done above for rubidium. For three-way parallelism, for example, we can schedule the rotations as

| | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 15 | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ |
| 14 | 4 | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ |
| 0 | 19 | 24 | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ |
| 13 | 17 | 22 | 26 | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ |
| 1 | 3 | 19 | 24 | 28 | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ |
| 12 | 2 | 17 | 21 | 26 | 30 | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ |
| 2 | 11 | 15 | 19 | 23 | 28 | 32 | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ |
| 11 | 6 | 13 | 16 | 21 | 26 | 30 | 34 | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ |
| 3 | 2 | 11 | 14 | 18 | 23 | 28 | 32 | 35 | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ |
| 10 | 1 | 10 | 13 | 16 | 21 | 25 | 30 | 33 | 36 | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ |
| 4 | 5 | 8 | 11 | 14 | 18 | 23 | 27 | 31 | 35 | 37 | $x$ | $x$ | $x$ | $x$ | $x$ | $x$ |
| 9 | 4 | 7 | 9 | 12 | 16 | 20 | 25 | 29 | 33 | 36 | 38 | $x$ | $x$ | $x$ | $x$ | $x$ |
| 5 | 3 | 6 | 8 | 10 | 14 | 18 | 22 | 27 | 31 | 34 | 37 | 39 | $x$ | $x$ | $x$ | $x$ |
| 8 | 1 | 5 | 7 | 9 | 12 | 15 | 20 | 25 | 29 | 33 | 36 | 38 | 40 | $x$ | $x$ | $x$ |
| 6 | 2 | 4 | 6 | 8 | 10 | 13 | 17 | 22 | 27 | 31 | 34 | 37 | 39 | 41 | $x$ | $x$ |
| 7 | 1 | 3 | 5 | 7 | 9 | 12 | 15 | 20 | 24 | 29 | 32 | 35 | 38 | 40 | 42 | $x$ |

TABLE II. Number of parallel steps (time) to synthesize a generic unitary operation $U$, up to a diagonal gate $D$, on a single atomic qudit. The number in parentheses is our best lower bound. Each Raman pair of laser beams counts as a single resource, and the first column indicates the number of pairs used in our $QR$ diagonalization of $U$. The tradeoff between time and resources is evident.

| Parallelism | Steps $^{87}$Rb ($d=8$) | $^{133}$Cs ($d=16$) |
|---|---|---|
| seven-way | 13 (11) | 29 (26) |
| six-way | 13 (11) | 30 (26) |
| five-way | 13 (11) | 31 (26) |
| four-way | 13 (11) | 35 (26) |
| three-way | 13 (11) | 42 (42) |
| two-way | 15 (15) | 62 (61) |
| one-way | 28 (28) | 120 (120) |

A summary of the tradeoff between resources and gate times with qudits encoded in ground hyperfine levels of $^{87}$Rb and $^{133}$Cs is given in Table II. As noted above, the two-way construction of 15 steps for $^{87}$Rb is optimal. Similar reasoning gives the two- and three-way lower bounds for $^{133}$Cs; for example, 118 rotations divided by 3 gives 40 steps plus two final steps for the last two rotations. The other lower bounds in the table are obtained assuming a completely connected coupling graph and $(d/2)$-way parallelism. In that case, if $d=2^p$, we can insert $d/2$ zeros in the first column at step 1, up to $d/4$ zeros in the first two columns 2 at step $2,\ldots,1$ zero in the first $p$ columns at step $p$, and then start the reduction in the $j$th column for $j=p+1,\ldots,d-1$ at step $\log_2 d+2(j-\log_2 d)$, for a total of $\log_2 d+2(d-1-\log_2 d)$ steps. Other choices of rotation sequences may reduce some entries in the table.

### D. Parallel diagonal gates

Up to this point our discussion has counted the number of parallel steps needed to construct any single-qudit unitary up to a diagonal gate $D$. Synthesizing the diagonal gate is unnecessary if the target qudit will remain dormant until a measurement in the computational basis. However, if the qudit will be targeted by subsequent operations then it will be necessary to phase the basis states of the qudit appropriately. We next consider parallel constructions for $D$. There are two variations of this problem to discuss. In the first, we define a gate to be an evolution by the generator $\lambda^z_{j,k}$, where $j$ and $k$ are paired levels. In the second, the gate library is restricted to Givens rotations [Eq. (3)] as is the case in systems controlled with on-resonant Raman laser pairs. Here one cannot realize a diagonal Hamiltonian directly but rather may simulate $e^{it\lambda^z_{j,k}}$ using a Euler angle decomposition.

First, note that the $D$ gate itself need only be simulated up to a local phase: e.g., we may chose $D\in \mathrm{SU}(d)$. Simulating a diagonal gate with $d-1$ independent phases should require appropriate couplings between $d-1$ pairs of states. There is a large amount of freedom in the choice of the set of the $d-1$ state pairs: any $D\in \mathrm{SU}(d)$ can be written $D=\prod_{m=1}^{d-1} e^{i\phi_{j_m,k_m}\lambda^z_{j_m,k_m}}$, provided the set of edges $E=\{(j_m,k_m)\}$ creates a spanning tree of the coupling graph. For $\{i\lambda^z_{j,k}:(j,k)\in E\}$ spans the diagonal subalgebra of $\mathfrak{su}(d)$, and therefore we may construct $\{\phi_{j_m,k_m}\}$ by solving $d-1$ linear equations [8]. Since diagonal gates commute, the simulation (in terms of $\lambda^z_{j,k}$) is maximally parallel, requiring one step. If only $k$-wise parallelism is allowed, then the number of steps is $\lceil (d-1)/k\rceil$.

We next consider the case that only $\lambda^x_{j,k}$ and $\lambda^y_{j,k}$ are allowed. Again choose any spanning tree for the coupling graph and construct $\{\phi_{j_m,k_m}\}$ by solving $d-1$ linear equations. Color the edges of the tree so that no node has two edges of the same color. (For example, in Fig. 3 we need three colors because node two has three edges.) Now for any edge $(j,k)$, we may indeed realize $e^{i\phi_{j,k}t\lambda^z_{j,k}}=e^{it_1(j,k)\lambda^x_{j,k}}e^{it_2(j,k)\lambda^y_{j,k}}e^{it_3(j,k)\lambda^x_{j,k}}$ for appropriate timings. Evolutions $e^{it_1\lambda^x_{j,k}}$ and $e^{it_2\lambda^y_{j,k}}$ do not commute and may not be applied in the same time step. Yet we may group the evolutions for a single color—black, for example—in three time steps as

$$\left(\prod_{(j,k)\ \mathrm{black}} e^{i\phi_{j,k}t_1(j,k)\lambda^x_{j,k}}\right)\left(\prod_{(j,k)\ \mathrm{black}} e^{i\phi_{j,k}t_2(j,k)\lambda^y_{j,k}}\right) \times\left(\prod_{(j,k)\ \mathrm{black}} e^{i\phi_{j,k}t_3(j,k)\lambda^x_{j,k}}\right). \tag{7}$$

Given a sufficient number of operations per step, this realizes $D$ in $3c$ parallel steps, where $c$ is the number of colors, regardless of the number of levels in the spanning tree. Hence, the construction is optimized by choosing a spanning tree that minimizes the number of colors. The number of colors $c$ is bounded by the maximum valency $c_m$ of any node in the coupling graph; if the coupling graph itself is a tree, then the number of colors is exactly $c_m$. When control resources are limited, we make a similar coloring, but limit the number of edges of a given color to the maximum number of operations allowed per step.

The spanning tree of Fig. 3 for $^{87}$Rb requires three colors for the edges. A diagonal computation can be done with the gate sequence $D=[e^{i\phi_{3,2}\lambda^z_{3,2}}][e^{i\phi_{2,5}\lambda^z_{2,5}}e^{i\phi_{0,7}\lambda^z_{0,7}}e^{i\phi_{4,1}\lambda^z_{4,1}}]\times[e^{i\phi_{5,0}\lambda^z_{5,0}}e^{i\phi_{2,4}\lambda^z_{2,4}}e^{i\phi_{1,6}\lambda^z_{1,6}}]$, which requires nine parallel Givens rotations, for each bracket in the expression for $D$ requires $x$, $y$, and $x$ steps. Similarly, cesium requires nine parallel Givens rotations.

The above treatment works for synthesizing an arbitrary diagonal gate $D$ without prior processing. However, generically, the gate $D$ follows the diagonalization process described in Sec. II C. In that case some pairwise phasing operations can be subsumed in earlier steps, therefore reducing the total number of pulse sequences. First, since Proposition 1 can be extended to any unitary, not just rotations of the form $G_{jk}$, we can apply a phase correction using edge $(j,k)$ as soon as we are finished with those two rows in the diagonalization. Second, we are allowed to choose an edge set for phasing different from the one we used for diagonalization. For example, using three-way parallelism for rubidium, at times 11,12, and 13 of the diagonalization, we can apply a phase correction using edge (0,6); at times 14, 15, and 16

we can use (7,0), (6,1), and (5,2); and at times 17, 18, and 19 we can finish by using (0,5), (1,4), and (2,3). A similar idea works for cesium using seven-way parallelism: at times 27, 28, and 29 use edge (0,14); at times 30, 31, and 32 use (15,0),(14,1),(13,2),(12,3),(11,4),(10,5), and (9,6); and at times 33,34, and 35 use (0,13), (1,12),(2,11),(3,10),(4,9),(5,8), and (6,7). In rubidium, an extra six Raman pulse sequences is optimal for phasing when nodes $|2\rangle$ and $|3\rangle$ are involved in the last rotation, and six pulses ending on $|6\rangle$ and $|7\rangle$ is optimal for cesium. We require no more than three and seven simultaneous couplings, respectively, which is also the number required for optimal diagonalization.

## III. PARALLELIZED NONLOCAL TWO-QUDIT GATES

In this section we propose an implementation of an arbitrary nonlocal unitary $U \in U(d^2)$ between two qudits $A$ and $B$. We suppose the qudits are spatially separated in some quantum-computing architecture, yet this architecture has (i) the capability to prepare a large reservoir of maximally entangled ($d=2$) qubits and (ii) the ability to shuttle halves of such Bell pairs so that they are spatially close to qudits $A$ and $B$. Hence, part of the costing is the number of such Bell pairs (e-bits) consumed in the nonlocal gate. To be clear, we describe only a nonlocal two-qudit gate rather than a teleported two-qudit gate meaning that quantum operations are performed on two qudits rather than four. The optimization of such a nonlocal gate presented here arises by considering its component rotations in terms of the $QR$ decomposition.

Before stating the protocol, we argue for why it is needed. Two criteria must be satisfied to realize high-performance two-qudit gates. First, nonlocality itself is desirable; most quantum-computer architectures impose spatial limitations on interqudit couplings. It is very inconvenient to simply accept this limitation, since fault-tolerant computation requires connectivity [17]. Now one might also suggest directly swapping qudits in order to achieve the required connectivity. Yet the swap gate itself may be faulty, and thus the resources required to make swapping fault tolerant might be prohibitive.

Second, reliable computation requires high-fidelity two-qudit gates. Usually, Hamiltonians capable of entangling distinct qudits are difficult to engineer (at any fidelity) and would require effort to optimize for fidelity. Thus, one would likely choose a particular physically available entangling two-qudit Hamiltonian, e.g., perhaps the controlled-phase gate $P_0=e^{i\pi|0\rangle\langle 0|\otimes|0\rangle\langle 0|}$, and then exploit this with local unitary similarity transforms to achieve arbitrary Givens rotations between qudit levels. The entire process might simulate any $U \in U(d^2)$ [8]. Local unitary similarity transforms arose naturally in this discussion, and this further implies that two-qudit nonlocality in such a scheme would follow, given a nonlocal protocol for a single entangling Hamiltonian.

It is difficult to design an architecture for two-qudit unitaries which allows for both high fidelity and high connectivity. Some possibilities are noteworthy. As opposed to a chain of swapping operations, distant qudits might be swapped using entanglement resources. Then a nonlocal gate between qudits $A$ and $B$ can be done by teleporting $A$ to a location neighboring $B$, performing an entangling gate between $A$ and $B$ and teleporting back. Typically, entangled qudits (e-dits) rather than e-bits are used to teleport qudits; i.e., each teleportation is performed with the assistance of a maximally entangled two-qudit resource $|\Phi_d^+\rangle=\frac{1}{\sqrt{d}}\Sigma_{j=0}^{d-1}|j\rangle|j\rangle$ [18]. While the amount of entanglement consumed using the resource $|\Phi_d^+\rangle$ is low, i.e., one e-dit=log($d$) e-bits, such a protocol would still require high fidelity (local) two-qudit gates between $A$ and $B$. As hinted at in the first paragraph of this section, a second alternative is to teleport the gate itself using an adaptation of the two-qubit gate teleportation protocol [19,20] [[21], Sec. II]. In such an implementation one would build a generic two qudit gate between $A$ and $B$ using multiple applications of a gate teleport sequence where each sequence consumed two e-dits. Such a protocol would require the preparation of high fidelity e-dits and the implementation of generalized two-qudit Bell-measurements between a memory qudit and one-half of an e-dit.

Here we describe a simple protocol for implementing a nonlocal two qudit gate, using high-fidelity e-bit resources. This has practical advantages over teleportation with e-dits because there exists advanced technology for producing high-fidelity "flying" photonic e-bits in the laboratory. Quantum memory might be stored in spatially separated qudits. A nonlocal gate between a pair could be realized by coupling each qudit to one-half of a polarization-entangled pair of photons, e.g., using hyperfine state-dependent coupling of an atom trapped in an optical cavity [22]. Furthermore, this scheme can be parallelized so that the entire implementation time for the nonlocal gate, as measured by the number of steps involving controlled rotation gates, is reduced by a factor of $O(d)$.

### A. A nonlocal controlled unitary gate

Consider a one-qudit unitary gate controlled on dit ($d-1$):

$$\wedge_1(V)=\sum_{j=0}^{d-2}|j\rangle\langle j|\otimes \mathbf{1}_d+|d-1\rangle\langle d-1|\otimes V.$$

We label the control qudit $A$ and the target qudit $B$. This subsection describes how such a gate can be implemented using:

(1) Operators local to $A$ and $B$.

(2) An e-bit. The ancilliary e-bit is encoded in a pair of qubits, say $A_1$ and $B_1$, again with $A_1$ neighboring $A$ and $B_1$ neighboring $B$. The joint state of the ancilla is the Bell pair $|\Phi^+\rangle=(1/\sqrt{2})\ (|00\rangle+|11\rangle)_{A_1,B_1}$.

(3) A controlled-NOT gate controlled on the qudit and targeting an ancilliary qubit. As a formula, this gate is $\wedge_1(\sigma^x)=\Sigma_{j=0}^{d-2}|j\rangle\langle j|\otimes \mathbf{1}_2-|d-1\rangle\langle d-1|\otimes\sigma^x$.

(4) A spatially local controlled $V$ gate with control an ancilla bit. As a formula, this is $\wedge_1(V)=|0\rangle\langle 0|\otimes \mathbf{1}_d+|1\rangle\langle 1|\otimes V$.

The controlled gate of item 3 should be considered to be primitive and highly engineered as discussed in the previous section. The controlled gate of item 4 might be decomposed

into local gates and the gate of item 3 using standard techniques [8–10].

The procedure for realizing $\wedge_1(V)$ is as follows.

(1) Apply $\wedge_1(\sigma^x_{A_1})$ with $A$ as control and $A_1$ as target.

(2) Measure $(\mathbf{1}_2+\sigma^z_{A_1})/2$. Send the one-bit (c-bit) classical measurement result $m_1$ to the side of qudit $B$.

(3) Perform $e^{i\pi m_1 \sigma^x_{B_1}/2}$ on the $B$ side of the architecture.

(4) Apply the operation $|0\rangle\langle 0|\otimes \mathbf{1}_d+|1\rangle\langle 1|\otimes V$ with $B_1$ as control and $B$ as target.

(5) Measure $(\mathbf{1}_2+\sigma^x_{B_1})/2$ and send the c-bit measurement result $m_2$ to $A$.

(6) Apply a relative phase to state $d-1$ of $A$ if and only if $m_2=1$, i.e., apply $P_{d-1}=e^{i\pi m_2|d-1\rangle_{AA}\langle d-1|}$.

### B. Bootstrap to nonlocal two-qudit state synthesis

We next consider the question of building a nonlocal two-qudit state-synthesis operator. We may write any two-qudit state $|\psi\rangle=\Sigma_{j=0}^{d-1}|j\rangle\otimes|\tilde{\psi}_j\rangle$, where the kets $|\tilde{\psi}_j\rangle$ are unnormalized. We also take the convention that $W|\psi\rangle=|0\rangle$ so that $W^\dagger|0\rangle=|\psi\rangle$. Using the partition of the state vector, one may show that any two-qudit state-synthesis operator $W$ can be decomposed into $d-1$ elementary controlled-rotation operators as follows [6]:

$$W=(V_d\otimes \mathbf{1}_d)\prod_{j=0}^{d-2}[(F_{d-1-j}\otimes \mathbf{1}_d)\wedge_1(V_{d-1-j})(F^\dagger_{d-1-j}\otimes \mathbf{1}_d)]\times(\mathbf{1}_d\otimes V_0). \tag{8}$$

Here we intend $F_j=|j\rangle\langle d-1|+|d-1\rangle\langle j|+\Sigma_{k\neq j,d-1}|k\rangle\langle k|$ to be a state-flip operator. The single-qudit operators $V_j$ are chosen so as to perform $V_j|\tilde{\psi}_j\rangle=r_j^{1/2}|0\rangle$, where $\langle\tilde{\psi}_j|\tilde{\psi}_j\rangle=r_j$ [8]. Then $V_0$ clears the remaining nonzero amplitudes.

The last subsection implicitly describes a nonlocal implementation of a controlled (one-qudit state synthesis) operator $W$, in that it details a scheme for the nonlocal $\wedge_1(V_{d-1-j})$. The resulting circuit for $W$ is shown in Fig. 8 and requires $d-1$ e-bits and $2(d-1)$ c-bits. Remarkably, the protocol can be parallelized to seven computational steps. Here by a single step we mean a set of operations that is no more time consuming than a controlled one-qudit rotation $\wedge_1(V)$, which itself can be decomposed into controlled-phase gates and single-qudit Givens rotations if so needed. The only nonobvious parallel step is step 4. Note that the operators $V_j$ generally do not commute. However, just before and just after this step, the usual teleportation case study shows that the state of the system lies within the span of those $|k\rangle=|k_0\rangle_A\otimes_{j=1}^{d-1}|k_j\rangle_{B_j}\otimes|k_d\rangle_B$ in which at most a single $k_j$ is one for $1\leq j\leq d-1$. Let $P$ denote the projection of Hilbert space onto the span of all $|k\rangle$ as above. If $Q$ denotes the central product of Eq. (8), we have

$$PQP=\prod_{j=1}^{d-1}e^{-it_j|1\rangle_{B_jB_j}\langle 1|\otimes h_j}, \tag{9}$$

for the map of Hamiltonians $h\mapsto PhP$ has image equal to the span of all $|1\rangle_{B_jB_j}\langle 1|\otimes h$. Moreover, for $j_1\neq j_2$ and any Hermitian $h_1,h_2$, we have $[|1\rangle_{B_{j_1}B_{j_1}}\langle 1|\otimes h_1,|1\rangle_{B_{j_2}B_{j_2}}\langle 1|\otimes h_2]=0$. Hence we can generate the gates in step 4 in parallel if we choose $t_j$ and $h_j$ such that $V_j=e^{-it_jh_j}$,. The operations in step 5 correspond to measurement of qubits $B_j$ in the Hadamard basis and count as a single parallel operation.

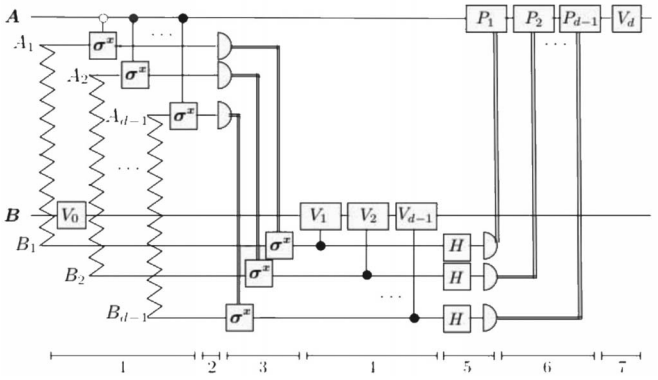


FIG. 8. A nonlocal two-qudit gate $U=W^\dagger$ that realizes the state synthesis $U|0\rangle_{A,B}=|\psi\rangle_{A,B}$ on qudits $A$ and $B$ using $d-1$ ancillary qubit pairs (indicated by sawtooth lines) each prepared in the state $|\phi^+\rangle_{A_j,B_j}=1/\sqrt{2}(|00\rangle+|11\rangle)_{A_j,B_j}$. Each qubit $A_j(B_j)$ in the entangled resource can constitute a new particle or a distinct degree of freedom of qudit $A(B)$. Controlled-NOT gates between $A$ and $A_j$ are conditioned on the basis state $|j\rangle_A$, as indicated by the shading of the control bubble. Double lines denote classical controlled operations dependent on qubit measurement outcomes, $H=e^{i\pi(\sigma^x+\sigma^z)/2\sqrt{2}}$, and $P_j=e^{i\pi|j\rangle\langle j|}$. The sequence of steps that can be implemented in parallel is indicated at the bottom.

### C. Spectral decomposition bootstrap to nonlocal gates

This protocol can be extended to implement an arbitrary nonlocal unitary $U\in U(d^2)$ between $A$ and $B$. Consider the spectral decomposition Eq. (4) $U=\Pi_{\ell=0}^{d^2-1}W_\ell C_\ell W_\ell$. Each controlled-phase operator $C_\ell$ is locally equivalent to the operator $\wedge_1[\mathbf{1}_d+(e^{i\phi}-1)|d-1\rangle\langle d-1|]$ and thus can be implemented in one step using one e-bit and two c-bits. Using our resource count for state-synthesis operators $W_\ell$ any two-qudit unitary can then be built using $\ell=7\times 2d^2+d^2=15d^2$ parallel operations with the assistance of $2\times(d-1)\times d^2+d^2=2d^3-d^2$ e-bits and twice that number of c-bits. To gauge the speedup, we note that a generic two-qudit operation without the assistance of ancillae requires $O(d^4)$ single-qudit operations and $O(d^3)$ controlled-unitary gates $\wedge_1(V)$ [8]. As measured by the number of sequential controlled-unitary gates, the circuit depth is then parallelized by a factor $O(d)$.

Recently, an alternative construction of two-qudit operations using qubit entanglement resources was proposed [23]. That work describes how a single e-bit and two c-bits suffice to implement a one-parameter subgroup of $U(d^2)$ between two distant qudits $A$ and $B$ with probability 1. Specifically, the protocol realizes unitaries of the form $V(\phi)=\exp[i\phi U_A\otimes U_B]$ where the operators $U_A,U_B$ are unitary and Hermitian. However, that work does not provide an algorithm for gen-

erating an arbitrary two-qudit unitary nor is there a count of the number of e-bits consumed in a covering of $U(d^2)$ with such unitaries.

### D. Improved fidelity by purification

The requirements for a successful implementation of the nonlocal gate described above are high-fidelity local unitaries and qudit-qubit operations. In practice, the most error-prone operations will be the latter because coupling two spatially distinct particles involves interactions mediated by a field which can also couple to the environment and thus decohere the system. A way to obviate this problem is to make the qubit degrees of freedom "local" to the qudits in such a way that those coupling operations can be made error tolerant. This can be accomplished by encoding a qudit $A(B)$ and the qubits $\{A_j\}(\{B_j\})$ in one and the same particle. That is, we chose to encode quantum information in particles each endowed with a tensor product structure $\mathcal{H}=\mathcal{H}_{\text{qudit}}\otimes\mathcal{H}_{\text{ancillae}}$ so that one subsystem is used to encode the qudit and the ancillary subsystem with state space $\mathcal{H}_{\text{ancillae}}=\otimes_j\mathcal{H}_{\text{qubit }j}$ is used to assist in two-qudit gate performance.

Dür and Briegel [24] showed that when the single-particle Hilbert space decomposes as $\mathcal{H}=\mathcal{H}_{\text{qubit}}\otimes\mathcal{H}_{\text{ancillae}}$ one can perform extremely high-fidelity two-qubit gates. In their protocol, information is encoded in a two-dimensional degree of freedom of each particle, say spin. Entanglement between particles is generated using coupling between ancillary degrees of freedom local to each particle such as quantized states of motion along $\hat{x},\hat{y}$, or $\hat{z}$. The prepared entanglement may not be perfect. Yet by using nested entanglement purification with two or more degrees of freedom, one can prepare a highly entangled state in the ancillary degrees of freedom with nonzero probability. If a purification round fails, then the entangled state can be prepared again without disturbing the quantum information encoded in the other (spin) degree of freedom. Given this, a nonlocal controlled-NOT gate can be implemented between the encoded qubits. This protocol is readily extended to nonlocal gates between qudits using the quantum circuit above. The only additional requirement is a degree of freedom that has a state space with dimension $d$. This could be accommodated using a truncated Fock space of a harmonically trapped particle or using a particle with internal structure having spin $(d-1)/2$. Gates between different degrees of freedom of the same particle, such as coupling spin to motion in trapped ions [25] or atoms [26], may be implemented with high precision using well-developed mechanisms for coherent control.

## IV. CONCLUSIONS

Quantum computation with qudits requires more control at the single-particle level than with qubits. It might be expected that the additional time needed to control all the levels would be prohibitively long in terms of memory decoherence times. We have shown how parallel (time-step optimized) one-qudit and two-qudit computation help surmount such difficulties. Given a qudit with a connected coupling graph, the time complexity for constructing an arbitrary unitary can be reduced at the expense of additional control resources. Even for systems with little connectivity between states, such as in the case of a qudit encoded in hyperfine levels of an alkali-metal atom, the number of parallel elementary gates can be made close to the optimal count for a maximally connected state space. For the purposes of two-qudit gates, we found a nonlocal implementation of an arbitrary unitary using $O(d^2)$ parallel steps. The protocol uses $O(d^3)$ e-bits which could in principle be prepared and distributed ahead of time with high fidelity.

Some outstanding issues remain. First, our treatment focused on systems with allowed couplings between pairs of states. In other systems, the selection rules may dictate a different set of subalgebras to be used for quantum control, e.g., spin-$(d-1)/2$ representations of the algebra $\mathfrak{su}(2)$. Some particular computations may be realized with much greater efficiency using such generators. Second, fault-tolerant computation relies not on exactly universal computation, but rather on approximating unitaries using a discrete set of one- and two-qudit gates. It would be worthwhile to investigate optimal protocols for implementing a discrete set of fault-tolerant nonlocal two-qudit gates using entangled qubit pairs.

## ACKNOWLEDGMENTS

The work of D.P.O. was supported in part by the National Science Foundation under Grants No. CCR-0204084 and No. CCF-0514213. G.K.B. received support from a DARPA/QUIST grant. We are grateful to Samir Khuller for providing Ref. [14].

[1] J. Preskill, Proc. R. Soc. London, Ser. A **454**, 385 (1998).
[2] C. Moore and M. Nilsson, SIAM J. Comput. **31**, 799 (2001).
[3] R. Raussendorf and H.-J. Briegel, Quantum Inf. Comput. **6**, 433 (2002).
[4] A. Yimsiriwattana and S. J. Lomonaco, e-print quant-ph/0403146.
[5] P. Hoyer, e-print quant-ph/9702028.
[6] S. S. Bullock, D. P. O'Leary, and G. K. Brennen, Phys. Rev. Lett. **94**, 230502 (2005).
[7] D. Gottesman, Chaos, Solitons Fractals **10**, 1749 (1999).
[8] G. K. Brennen, D. P. O'Leary, and S. S. Bullock, Phys. Rev. A **71**, 052318 (2005).
[9] A. Muthukrishnan and C. R. Stroud Jr., Phys. Rev. A **62**, 052309 (2000).
[10] E. Knill, e-print quant-ph/9508006.
[11] F. Albertini and D. D'Allesandro, Linear Algebr. Appl. **350**, 213 (2002).
[12] I. H. Deutsch and P. S. Jessen, Phys. Rev. A **57**, 1972 (1998).
[13] D. P. O'Leary and S. S. Bullock, Electron. Trans. Numer. Anal. **21**, 20 (2005).

[14] Xin He and Yaacov Yesha, J. Algorithms **9**, 92 (1988).
[15] Note that this transition corresponds to $|F_{\downarrow},M_F=0\rangle \rightarrow |F_{\uparrow},M_F=0\rangle$. Selection rules for Raman laser pulses allow this transition only if both the pump and probe pulses are $\sigma_+$ or $\sigma_-$ polarized. Because this particular transition is not needed for parallelism we can fix the pump or probe pulse to have $\pi$ polarization and still access all the other couplings in the graph using frequency selectivity.
[16] D. P. O'Leary and G. W. Stewart, Linear Algebr. Appl. **77**, 275 (1986).
[17] M. Oskin, F. T. Chong, and I. L. Chuang, Computer **35**, 79 (2002).
[18] C. H. Bennett *et al.*, Phys. Rev. Lett. **70**, 1895 (1993).
[19] D. Gottesman and I. L. Chuang, Nature (London) **402**, 390 (1999).
[20] J. Eisert *et al.*, Phys. Rev. A **62**, 052317 (2000).
[21] R. J. Jozsa, e-print quant-ph/0508124.
[22] C. Monroe, Nature (London) **416**, 238 (2002).
[23] H.-S. Zeng, Y.-G. Shan, J-J Nie, and L.-M. Kuang, e-print quant-ph/0508054.
[24] W. Dür and H.-J. Briegel, Phys. Rev. Lett. **90**, 067901 (2003).
[25] C. A. Sackett *et al.*, Nature (London) **404**, 256 (2000).
[26] D. J. Haycock *et al.*, Phys. Rev. Lett. **85**, 3365 (2000).