Logical Majorana fermions for fault-tolerant quantum simulation

Andrew J. Landahl\textsuperscript{1,2,3,} \textasteriskcentered{} and Benjamin C. A. Morrison\textsuperscript{1,2,3,} \textdagger{}

\textsuperscript{1}Center for Computing Research, Sandia National Laboratories, Albuquerque, NM, 87185, USA
\textsuperscript{2}Center for Quantum Information and Control, University of New Mexico, Albuquerque, NM, 87131, USA
\textsuperscript{3}Department of Physics and Astronomy, University of New Mexico, Albuquerque, NM, 87131, USA

We show how to absorb fermionic quantum simulation’s expensive fermion-to-qubit mapping overhead into the overhead already incurred by surface-code-based fault-tolerant quantum computing. The key idea is to process information in surface-code twist defects, which behave like logical Majorana fermions. Our approach implements a universal set of fault-tolerant gates on these logical Majorana fermions by effecting encoded measurement-based topological quantum computing with them. A critical feature of our approach is the use of code deformations between logical tetron and logical hexon surface-code-patch encodings, which enables one to move beyond the limitations of a wholly square-patch tetronic surface-code approach. To motivate near-term implementations, we also show how one could realize each of a universal set of logical Majorana gates on a small-scale testbed using noisy intermediate scale quantum (NISQ) technology on as few as 13 qubits.

I. INTRODUCTION

Quantum computers are expected to excel at simulating quantum systems, taking us far beyond the reach of conventional computers\textsuperscript{[1–3]}. In fields like quantum chemistry, materials science, nuclear physics, and high-energy physics, these systems are frequently comprised of fermions\textsuperscript{[4, 5]}. If the quantum computer simulating them uses qubits to store quantum information, then there is an unavoidable and substantial overhead required to map the fermions to the qubits\textsuperscript{[6–15]}. Moreover, these qubits must have very low error rates for the results to be trustworthy. This means that the relevant qubits will likely need to be logical qubits that are realized by quantum error correcting codes, and these codes must be processed using fault-tolerant quantum computing protocols, which adds even more overhead.

Here we ask and answer the question, “Is it possible to use a quantum computer that processes logical fermions instead, thereby bypassing the overhead of mapping fermions to logical qubits?” In order to do that, one would need (a) a model of universal fermionic quantum computation, (b) a method for constructing arbitrarily reliable logical fermions using physical qubits, and (c) fault-tolerant protocols for realizing universal fermionic quantum computation with those logical fermions. The first two of these have already been developed\textsuperscript{[7, 16]}; here we show how to realize the third. Consequently, we show that the overheads incurred by fermion-to-qubit mappings are only relevant for direct implementations by noisy intermediate scale quantum (NISQ) devices. This overhead will conveniently vanish in the awaited fault-tolerant application scale quantum (FASQ) era, although small-scale demonstrations of this approach should be possible with NISQ technology.

The remainder of this paper is organized as follows. In Sec. II, we review the pertinent background on fermionic quantum circuits and logical fermion codes. In Sec. III, we describe how to realize a complete set of universal operations for fermionic quantum circuits on logical fermions realized by qubit-based surface-code patches. In Sec. IV, we present examples of each of these logical operations on small surface codes that could be realized using NISQ technologies. In Sec. V, we conclude.

II. BACKGROUND

A. Majorana fermions and fermionic Hamiltonians

Consider a collection of fermions obeying the following canonical anticommutation relations on their elementary creation ($f^\dagger$) and annihilation ($f$) operators:

\[
\{f_p, f_q^\dagger\} = \delta_{pq}, \quad \{f_p, f_q\} = \{f_p^\dagger, f_q\} = 0. \tag{1}
\]

Because these relations discriminate between particles and antiparticles, we call them Dirac fermions.

One can always formally split a Dirac fermion into a pair of Majorana fermions, either of which does not discriminate between particles and antiparticles. The corresponding elementary Majorana fermion operators are

\[
c_{2p} := f^\dagger + f, \quad c_{2p+1} := i(f^\dagger - f). \tag{2}
\]

The induced Majorana fermion relations are

\[
\{c_p, c_q\} = 2\delta_{pq}, \quad c_p^\dagger = c_p. \tag{3}
\]

Convenient derived fermionic operators include the mode number operator,

\[
n_p := f_p^\dagger f_p = ic_{2p}c_{2p+1}, \tag{4}
\]

and the total mode number operator,

\[
n := \sum_p n_p. \tag{5}
\]
Fermionic Hilbert space, an example of a Fock space, is the completion of the infinite direct sum of antisymmetrized eigenspaces of \( n \). Its standard basis is the set of Fock states, which are states having definite eigenvalues for all \( n_p \). The eigenvalues \( N_p \) for \( n_p \) are restricted to be 0 or 1 by the commutation relations, while the eigenvalue \( N \) for \( n \) can take on any non-negative integer value, or even be countably infinite.

A local fermionic Hamiltonian is one that can be expressed as a sum of terms, each of which acts on a constant number of fermionic modes. Consequently, no term can contain an odd number of elementary fermionic operators, because such terms act nonlocally on fermionic Fock states:

\[
\begin{align*}
    f_p | \ldots, N_p, \ldots \rangle &= (-1)^{N_p} f_p | \ldots, N_p - 1, \ldots \rangle \quad (6) \\
    f_p^\dagger | \ldots, N_p, \ldots \rangle &= (-1)^{N_p} (1 - N_p) f_p^\dagger | \ldots, N_p + 1, \ldots \rangle. \quad (7)
\end{align*}
\]

A final fermionic operator worth noting is the total fermionic parity operator,

\[
Q := (-1)^n. \quad (8)
\]

Because each term in a local Hamiltonian \( H \) acts on an even number of modes, it necessarily obeys \([H, Q] = 0\), conserving total fermionic parity. However, it might might not also obey \([H, n] = 0\), conserving total mode number, as is the case for Hamiltonians that include BCS-like interactions [17, 18] of the form

\[
\sum_{p,q} (f_p f_q + f_q^\dagger f_p^\dagger). \quad (9)
\]

In terms of Majorana fermion operators, the most general form of an \( N \)-mode local fermionic Hamiltonian with only one-body and two-body interactions is

\[
H = \sum_{p,q=0}^{2N-1} g_{pq} c_p c_q + \sum_{p,q,r,s=0}^{2N-1} g_{pqrs} c_p c_q c_r c_s, \quad (10)
\]

where the \( g_{pq} \) and \( g_{pqrs} \) are real coefficients. This is the class of fermionic Hamiltonians we will show how to simulate fault-tolerantly.

**B. Majorana fermionic quantum circuits**

In the standard Majorana fermionic quantum circuit (MFQC) model of quantum computation [7], solving a computational problem is a three-step process:

1. Select a circuit from a \( P \)-uniform family of Majorana fermionic quantum circuits (classically) [19].
2. Execute the MFQC (quantumly).
3. Return the classical result, say, as a bit string (classically).

In this process, each MFQC in the family is expressible as a sequence of elementary operations on a collection of Majorana fermions. The elementary operations include preparations, measurements, and quantum coherent operations. Here, as is common, we restrict our attention to circuits in which each elementary operation acts on only a constant number of operands. If the set of elementary operations can be used to approximate any transformation on the Majorana fermions arbitrarily well by a sufficiently long circuit, then the set of operations is said to comprise a universal gate set. (We use the term “gate set” even when the set contains not just coherent gates but also preparation and measurement operations.)

In their landmark paper defining the standard MFQC model [7], Bravyi and Kitaev presented several universal gate sets. Li later presented a variant of one of these that is comprised mostly of preparation and measurement operations [20]. Li’s universal gate set consists of the following operations, for a collection of Majorana fermions indexed by the variables \( p, q, r, s, \) and \( s \):

1. Prepare a +1 eigenstate of the Majorana fusion operator, sometimes also called the Majorana exchange operator, \( F_{pq} = ic_p c_q \).
2. Measure the two-fermion observable \( F_{pq} \) (either destructively or non-destructively).\(^1\)
3. Measure the four-fermion observable \( F_{pq} F_{rs} \) non-destructively.
4. Prepare the “magic state” \( |T\rangle_{pqrs} \), which is the +1 eigenstate of the following observables:

\[
\frac{F_{pq} F_{rs}}{\sqrt{2}}. \quad (11)
\]

5. Apply the (unitary) Majorana exchange gate \( F_{pq} \).

As an aside, we note that because Majorana fermions realize a theory of anyons known as Ising anyons, it is impossible to measure them individually even if one wanted to—only measurements of groupings of them can reveal their collective anyonic charge. A −1 outcome obtained upon measurement of \( F_{pq} \) on a pair of Majorana fermions in modes \( p \) and \( q \) indicates that the two \( \sigma \) anyons used to represent them in the Ising anyon theory have fused to create the nontrivial \( \psi \) particle as opposed to the trivial (vacuum) \( 1 \) particle. For more details on this connection to Ising anyons, see, for example, the textbooks by Wang [22] and Pachos [23].

Fig. 1 depicts an example of a Majorana fermion circuit that implements the coherent Majorana exchange

\(^1\) By “non-destructive,” we mean that the post-measured state is an eigenstate of the observable measured, as per the standard von Neumann prescription [21].
gate $F_{ps}$ using only non-destructive two-fermion measurements. The circuit represents the scenario in which each of the measurements return the eigenvalue +1, but generally either +1 or −1 eigenvalues will be obtained. As noted by Bonderson et al. [24], one can correct any undesired outcome using a “forced measurement” protocol. More robustly, as noted by Zheng et al. [25], one can simply track the “Majorana frame” classically and re-interpret future measurement outcomes accordingly. This latter approach is similar to Pauli frame tracking used for qubit circuits [26, 27]. Indeed, using the analogue of Clifford frame tracking for qubits [28, 29], one can dispense with the final coherent gate in the Li universal gate set entirely by enacting “Majorana exchange frame” tracking [25]. In this way, the gate set realizes a model of measurement-based topological quantum computation described, e.g., in Refs. [24, 25, 30, 31].

![Fermionic Quantum Circuit](image)

**C. Majorana fermion stabilizer codes**

A Majorana fermion stabilizer code [32], or Majorana stabilizer code for brevity, is the simultaneous +1 eigenspace of a collection of commuting, Hermitian, even-weight Majorana operators. The evenness constraint ensures that these operators are fermion-parity preserving, and hence physically observable. Following the language used for qubit stabilizer codes [33], we say these operators generate the code’s stabilizer group, and each operator is called a stabilizer or a check for brevity, because they “stabilize” the codespace and are what are measured to “check” for errors. More generally, in a subsystem Majorana stabilizer code, the measured checks need not commute and the stabilizer group is defined to be the center of the check group. Whether for subspace or subsystem Majorana stabilizer codes, the logical group is the check group’s normalizer.

Without loss of generality, each check in a Majorana stabilizer code can be written as $S_\sigma$, where $\sigma$ indicates the set of modes $V_\sigma$ on which it has support. Each logical operator $L$ is supported on a set of modes $V_L$ obeying $|V_L \cap V_\sigma| \equiv 0 \mod 2$ for each $\sigma$ to ensure that $[L, S_\sigma] = 0$. Mathematically, each check and logical operator can be expressed as

$$S_\sigma := i^{|V_\sigma|/2} \prod_{q \in V_\sigma} c_q$$

$$L := \eta_L \prod_{q \in V_L} c_q,$$

where $\eta_L \in \{\pm 1, \pm i\}$ is a phase.

The distance of a Majorana fermion stabilizer code is the minimum nonzero weight of its logical group’s elements. Unlike qubit codes, a Majorana fermion stabilizer code of distance $d$ cannot necessarily correct all Majorana errors of weight $\lfloor (d - 1)/2 \rfloor$, because the codes treat even and odd logical operators on the same footing, exposing them, e.g., to “quasiparticle poisoning” errors from weight-one Majorana fermion operators, which do not conserve fermionic parity locally, but might do so when the environmental degrees of freedom of the bath are taken into consideration [34].

A fermion stabilizer code is a Majorana stabilizer code on an even number $(2n)$ of Majorana fermions [32, 34]. One can show that a fermion stabilizer code’s logical group is isomorphic to the Pauli group on $k = n - n_{\text{checks}}$ logical qubits, so that one can think of such codes as encoding $k$ logical qubits in $2n$ physical Majorana fermions [32]. Following Ref. [34], we use $[n, k, d]_f$, or alternatively $[2n, k, d]_m$, to denote a fermion stabilizer code that encodes $k$ qubits to distance $d$ in $n$ Dirac fermions, or equivalently, in $2n$ Majorana fermions.

A Majorana surface code is a Majorana stabilizer code defined by an embedding of a graph into a surface. Generally, one can use a rotation system to define such codes [35]. Here, we only need to consider a subclass of Majorana surface codes, first described by Litinski and van Oppen [36], that are defined by face-three-colorable graphs embedded on a disk in which one associates Majorana fermions with vertices and checks with faces. (The “face” outside the graph in the disk might not be able to be consistently colored with all the other faces.) In fact, we only will need to consider two simple versions of these codes, called the tetron code and the hexon code, described in the next section.

**D. Tetron and hexon codes**

Two simple examples of fermion surface codes are the tetron code and the hexon code, corresponding to the cycle graph embedded on the disk in which the cycle graph
has four or six vertices respectively [36–38]. (More generally, $2n$-on codes are fermion surface codes on cycle graphs of length $2n$ embedded on the disk.) In quantum coding notation, these simple examples are denoted by $[2, 1, 2]_f$ and $[3, 2, 2]_f$, respectively. Each of these codes has a single check, corresponding to the product of all of the Majorana fermion operators on the vertices, up to a phase consistent with Eq. (13). The constraint that the codes are $+1$ eigenstates of this check operator is just the even-parity constraint required of any collection of indistinguishable fermions generated by a local Hamiltonian. Consequently, the logical Majorana fermions in any such code block are indistinguishable from each other, but are distinguishable from logical Majorana fermions that may exist in other such code blocks.

There are many ways of choosing a basis for the logical Pauli operators for these codes. For the tetron code, we choose the logical operators

$$Z = ic_0c_1 \quad X = ic_1c_2.$$  

(15)

For the hexon code, we choose the logical operators

$$Z_1 = ic_0c_1 \quad \bar{X}_1 = ic_1c_2$$  

(16)

$$Z_2 = ic_4c_5 \quad \bar{X}_2 = ic_3c_4.$$  

(17)

These choices are captured in Fig. 2.

![Fig. 2: (Color online.) Examples of some simple Majorana surface codes.](image)

**E. Logical Majorana codes**

Like Majorana surface codes, qubit surface codes [39–46] can be defined using a rotation system that describes a graph embedding combinatorially [35]. As with Majorana surface codes, we will only consider a narrow subclass of such codes, namely those defined by face-two-colorable graphs containing squares and digons embedded on a disk in which one associates qubits with vertices and checks with faces. (The “face” outside the graph in the disk might not be able to be consistently colored with all the other faces.) The checks for one face color can be associated with a tensor product of Pauli $X$ operators on a face’s incident qubits and the checks for the other face color can be associated with a tensor product of Pauli $Z$ operators on a face’s incident qubits. Alternatively, all checks can be given the same local structure by a set of local basis changes, turning it into a so-called $XZZX$ code [47]. Because these codes are embedded on a disk, we call them qubit surface-code patches.

We will only need to consider two families of qubit surface code patches, the tetron and hexon families, in which the graphs are portions of a square lattice with digons around the perimeter alternating in such a way that there are only four or six disconnected external boundary colors, as depicted in Fig. 3. Topological quantum codes that have the property that the boundary color is permitted to change, or in which the face coloring cannot be maintained consistently internally, have been explored in multiple contexts [16, 48–57]. Borrowing the language of Ref. [16], we will call the defects at the transition points for the tetron and hexon qubit surface code families twist defects, or just twists, for brevity.

The similarity between the codes in Fig. 2 and Fig. 3 is not mere coincidence. The twist defects along the boundaries of these codes act like logical Majorana fermions (or, more precisely, like $Z_2$-crossed braided tensor categories [56]), as pointed out in Refs. [16, 52, 53]. In other words, these codes essentially encode logical Majorana fermions in physical qubits. To disambiguate from the term “Majorana fermion code,” which describes a code that works the other way around by encoding physical Majorana fermions into logical qubits [20, 32, 34, 36, 38, 40, 58–60], we will call these logical Majorana fermion codes, or just logical Majorana codes, for short.

Upon closer examination, one can see that the logical Majorana operators for the tetron and hexon qubit surface codes are themselves further encoded in the tetron and hexon codes, respectively. These qubit surface-code patches are therefore concatenated codes, in which physical qubits realize logical Majorana fermions, which in turn realize logical qubits. A helpful consequence is that quasiparticle poisoning errors are not a concern, because errors on physical qubit-level operations, once decoded with the lower-level logical Majorana code, translate at worst to even-weight logical Majorana operators in the upper-level tetron or hexon code. This means that a distance-$d$ qubit tetron or hexon surface code protects its encoded logical Majorana fermions against all physical errors of weight $\lfloor (d - 1)/2 \rfloor$ or less.

![Fig. 3: (Color online.) Distance-five versions of surface code patches that realize logical Majorana fermions as “twist defects,” depicted by white circles with red boundaries.](image)

If one wishes to consider the logical Majorana fermions
in qubit surface-code patches independently of the Majorana code that they form, one must consider arbitrary operators that act on \textit{collections} of such code patches. These do not necessarily correspond to logical Pauli operators for either code, and may have no meaning when one only narrowly thinks about surface codes as just encoding logical qubits. Formulating fault-tolerant protocols for operations related to these non-logical-qubit operators is the main contribution of this paper.

F. Universal qubit surface-code fault tolerance

While many sets of universal gates for qubit circuits are known, we will frequently appeal to the following one, because it is closely related to Li’s universal gate set for the MFQC model of quantum computation. The operations in this set apply to a collection of qubits indexed by the variables \( j \) and \( k \):

1. Prepare a +1 eigenstate of the qubit Pauli operators \( Z_j \) and \( X_j \).
2. Measure the qubit Pauli operators \( Z_j \) and \( X_j \) (either destructively or non-destructively).
3. Measure the two-qubit Pauli operators \( P_j \otimes P_k' \) non-destructively, where \( P_j \) and \( P_k' \) are each independently one of the Pauli operators \( X, Y, \) or \( Z \).
4. Prepare the “magic state” \( |T\rangle_j \) which is the +1 eigenstate of the observable

\[
\frac{X_j + Y_j}{\sqrt{2}}.
\]

5. Apply the (unitary) phase gate \( S_j = \sqrt{Z_j} \).

As with the Li universal gate set, the last gate is unnecessary and can be handled by Clifford-frame tracking [28].

III. RESULTS

Li’s universal gate set for Majorana fermionic quantum circuits described in Sec. IIB is universal for \textit{free} Majorana fermions, but the logical Majorana fermions stored in tetron qubit surface-code patches are \textit{bound} together. How can we realize the complete Li universal gate set on these bound fermions? Some surface-code-patch logical-qubit operations readily translate to logical Majorana operations on the bound fermions, but some operations in the Li universal gate set cannot be captured in this way. Establishing protocols for these new operations that have no direct analogues as logical-qubit operations is the focus of our attention.

To begin, it is helpful to divide the elements of the Li universal gate set into operations that are “on patch,” meaning they apply to fermions that lie on the same tetron qubit surface code, from those that are “between patches,” meaning that they apply to a collection of fermions that are distributed across two different tetron qubit surface codes. A set of operations that suffices to establish that the Li universal gate set can be realized using tetron qubit surface code patches is the following:

1. On-patch pair creation.
2. On-patch pair fusion.
3. On-patch non-destructive four-fermion measurement.
4. On-patch magic-state preparation.
5. On-patch exchange.
6. Between-patch exchange.

The reason that this set is sufficient is because, using operation 6, one can move any pair or quad of fermions to the same patch, apply any desired sequence of operations from 1–5, and then use operation 6 to move the fermions back to where they started.

To realize this set, it is helpful to utilize the following auxiliary logical fermionic operation:

7. Between-patch non-destructive four-fermion measurement of neighboring fermions.

In the context of the logical qubits stored in two neighboring tetron qubit surface code patches, operation 7 is called lattice surgery, and it has been studied in great detail, for example in Refs. [52, 61–64]. It is helpful to break this operation up into two steps: merging (fusion) and splitting (fission). In the merging step, one measures the collection of checks between the two patches (repeatedly, for fault tolerance) and uses their (decoded) product to determine the measured eigenvalue of the observable. When viewed as storing a pair of logical qubits, one interprets this as a fault-tolerant measurement of a Pauli product operator on the two patches. When viewed as storing logical Majorana fermions, with fermions \( p, q, r, \) and \( s \) being involved in the fusion, one interprets this as a fault-tolerant measurement of the logical \( F_{pq}F_{rs} \) operator, as noted by Brown \textit{et al.} [52]. Once merged, each patch is no longer subject to its own separate tetron-code fermion-parity constraint; instead, only a global even-parity constraint holds sway, expressible as the product of the previous two tetron stabilizer generators. The merging operation has fused two tetrions into a single tetron.

After just the fusion step, it is not clear whether to characterize the measurement as destructive or non-destructive, because two separate code blocks have merged into one. In the splitting step, the two separate patches are restored, making the characterization easier. During splitting, the checks fusing the patches together cease being measured (repeatedly, for fault tolerance), and the digon (half-moon) checks between them...
begin to be measured again. Although the fusion process that merges pairs of digons yields predetermined values when measured, they can be split with two different outcomes, determined at random, as depicted in Fig. 4. This randomness is important, because it ensures that only the total fermionic parity of all the fermions from both patches is preserved, while the fermionic parity of each individual tetron patch can change from what its original value was before the fusion. If the tetrons were guaranteed to go back to their separate even-parity fermionic sectors, they would have “forgotten” about each other and the interaction they had in the fusion process. Although the parities of the individual tetron patches may have changed, this should not be interpreted as an error. Rather, the new parity simply sets the new codespace eigenvalue that must be preserved by future error correction, as is measured by operation 3. In this sense, the two patches again encode separate tetrons, although they are entangled by the measurement of the four-fermion operator. The measurement is therefore non-destructive, because the two separate tetrons have not been lost.

With this auxiliary operation in hand, we cover protocols for realizing the first six operations in the next two sections.

A. On-patch operations

We consider each of the five on-patch operations described in the previous section in turn.

Operation 1: On-patch pair creation

Without loss of generality, using operations 5, one can move any two logical Majorana fermions on the patch so that they lie along the side of a surface-code qubit tetron patch on which the logical Pauli Z operator is supported. A fault-tolerant protocol for preparing a +1 eigenstate of the logical Z operator, such as protocols described in Refs. [65, 66], also prepares the +1 eigenstate of the corresponding logical Majorana fusion operator. In other words, any of these well known fault-tolerant logical Z preparation protocols also prepares a pair of logical Majorana fermions fault tolerantly.

Operation 2: On-patch pair fusion

As with operation 1, without loss of generality, using operation 5, one can move any two logical Majorana fermions on the patch so that they lie along the side of a surface-code qubit tetron patch on which the logical Pauli Z operator is supported. A fault-tolerant protocol for measuring the logical Z operator destructively, such as measuring all of the qubits in the Z basis and decoding the classical outcomes, also measures the corresponding logical Majorana fusion operator destructively.

By fault-tolerantly preparing an ancillary patch in the +1 eigenstate of logical Z, performing fault-tolerant lattice surgery between it and the patch so as to measure logical XX, and then fault-tolerantly measuring logical Z on the ancilla patch, one can alternatively measure the logical Majorana fusion operator non-destructively instead.

Operation 3: On-patch non-destructive four-fermion measurement

The four-fermion measurement on a patch corresponds to measuring the product of all checks in the surface code, which projects it onto the codespace. This is because the four-fermion measurement corresponds to the single stabilizer generator for the tetron code. If the qubits are already in the codespace, say, from a previous fault-tolerant operation, then there is no need to do this measurement at all. Whether the eigenvalue to be preserved is +1 or −1 depends on context. At the very beginning of a quantum computation, the value to be preserved is +1. However, after lattice surgery, as described earlier, it may be the case that the value −1 needs to be preserved.

Operation 4: On-patch magic-state preparation

Fault-tolerant protocols for injecting the magic state \((|0⟩ + e^{i\pi/4}|1⟩)/\sqrt{2}\) into a surface-code patch, such as those described in Refs. [62, 63, 65–67], also serve as fault-tolerant protocols for injecting the analogous Majorana-fermion magic state on the patch, as shown by Li [68].

Operation 5: On-patch exchange

Brown and Roberts [54] showed how to fault-tolerantly permute any pair of neighboring logical Majorana fermions on a patch, either clockwise or counterclockwise, by introducing an auxiliary surface code patch and
performing a sequence of lattice surgery operations between the original patch and the ancilla patch. A similar alternative approach is described by Litinski [29]. Either of these approaches allows one to exchange neighboring logical Majorana fermions on a surface-code qubit tetron patch.

One way of interpreting the on-patch exchange is as a phase ($S$) gate on the logical qubit stored on the patch. The quantum circuit that implements this via lattice surgery operations is depicted in Fig. 5.

![Fig. 5: (Color online.) Measurement-based circuit implementing the $S$ gate. The final Pauli correction $X^a Z^b$ can be tracked classically via the Pauli frame.](image)

**B. Between-patch Majorana exchange**

Unlike the on-patch operations, or even the between-patch non-destructive four-fermion measurement operation, the between-patch Majorana exchange operation is not automatically inherited from well-trodden fault-tolerant logical qubit operations on tetron qubit surface code patches. To realize this operation, we perform a reversible code deformation between two tetrions and a hexon. Unlike lattice surgery, no logical quantum information is lost in this reversible process. From the logical qubit perspective, this is a transformation between two codes that each hold one logical qubit to a code that holds two logical qubits. From the logical Majorana fermion perspective, this transformation is more subtle. Because each tetron holds four logical Majorana fermions, and because each is subject to a collective fermionic parity constraint, only two logical Majorana fermions per tetron are actually available to store useful information. When the two tetrions are fused into a hexon, again the collective fermionic parity constraint on the hexon limits the number of usable logical Majorana fermions, this time from six down to four. This is just enough to ensure that the logical Majorana fermionic information transfers reversibly from the two tetrions to the hexon.

Once the two tetrions are merged into a hexon, the on-patch operations described in the previous section, which are readily generalized from a tetron patch to a hexon patch, can be used to exchange the desired pair of logical Majorana fermions. Because the code deformation is reversible, the two tetrions can then be restored, completing the desired Majorana exchange between the tetrions.

There are multiple ways to fuse two tetrions into a hexon. Fig. 6 depicts a way to fuse a tetron and a mirrored tetron into a hexon by stretching the logical operators on both so that something similar, but not exactly, like logical $Y \otimes Y$ measurement will fuse them. The reason this is not a $Y \otimes Y$ measurement is that the lattice surgery only extends along half of the length of the logical $Y$ operators, so no information about either $Y$ operator is gleaned from the surgery.

A more compact way to realize the fusion is depicted in Fig. 7. In this figure, both tetrions start with the same handedness, but one is rotated relative to the other. The rotation can be implemented via standard protocols for surface-code patches, such as those described in Ref. [29]. The lattice surgery itself appears reminiscent of how a logical $CNOT$ gate is performed via lattice surgery [62, 63]. However, because of the patch rotation and lattice offsets, the fusion maps the two tetrions into a hexon.

These two constructions for between-patch Majorana exchange, which deform between two tetrions and a hexon, are the key to enabling direct processing of logical Majorana fermions stored in surface-code patches. Neither construction can be explained using the language of logical qubits isolated to individual surface-code patches, because the pairs of Majorana fermions used to define some of the logical operators necessarily become split across two tetrions during the constructions.

**IV. NISQ TESTBED**

The elimination of fermion-to-qubit mappings we have described are primarily relevant for fermionic simulation algorithms implemented on FASQ-era technology. However, it should be possible to demonstrate examples of each of the elementary FASQ operations required using...
existing NISQ-era testbed technology. We sketch how to do this here, assuming one has a NISQ-era quantum computer with as few as 13 qubits, assuming the same syndrome qubit is used to measure all of the checks (which does not scale fault-tolerantly). If the quantum processing is localized around the checks, then 23 qubits will suffice. We do not claim that these constructions are minimal in any sense, only that they can be carried out on NISQ-era machines of these modest sizes.

The code distances used in our examples here are all for $d = 2$, which means that single faults can be detected, but they cannot be corrected. This leads to a post-selected implementation, where errors are suppressed quadratically only on instances in which no errors are detected. Others have considered using standard $d = 2$ quantum error-detecting codes in conjunction with fermionic simulations, for example Urbanek et al. [69] considered wrapping a chemical calculation with an error-detecting $[4, 2, 2]$ code. However, our approach leaves the information carriers as logical Majorana fermions that can be manipulated with logical gates that act directly on the fermions. We have left our presentation in this section generic, emphasizing the elementary operations one could realize to demonstrate a universal set, rather than narrowly focusing on any specific fermionic simulation or any specific realization in a quantum computing technology, because our goal is to present the tools and methods one could use in a variety of NISQ testbed demonstrations of the approach.

### FIG. 7: (Color online.) Compact code deformation mapping two tetron qubit surface codes into one hexon qubit surface code.

(a) Two tetrions, offset. (b) Fuse. (c) Shrink.

### FIG. 8: (Color online.) On-patch pair creation. The pair of Majorana fermions created are on the right two corners (or left two, up to multiplication by stabilizer generators).

#### A. On-patch operations

**Operation 1: On-patch pair creation**

The $+1$ state is quite simple to prepare. Each of the four qubits that will be in the patch is prepared in the $|0\rangle$ state, and then the three checks are measured to project the patch into the code space. The three checks are then measured a second time, detecting errors by comparison with the first measurements. If there is a discrepancy, then an error has been detected.

**Operation 2: On-patch pair fusion**

On-patch pair fusion is facilitated by lattice surgery (operation 7) between the patch of interest and a patch prepared in a known state (using operation 1). As indicated in Fig. 9, checks are measured for two total rounds in step 9(a), and the state is discarded if the syndromes disagree. The fusion in step 9(b) is repeated for two total rounds, and again the state is discarded if the syndromes disagree. Finally, in step 9(c), the right four qubits are measured in the $Z$ basis while the checks are measured one more time and checked for disagreement with their previous outcomes. The $Z$ basis measurement does not need to be repeated, but classical error detection is done on the outcomes of the measurements to validate that their total parity is even.

**Operation 3: On-patch non-destructive four-fermion measurement**

Measuring the four fermions on a patch non-destructively is as simple as the pair creation process in operation 1: one simply measures the check operators two times, and if the syndromes differ, and error has been detected. The value of the product of the checks is the value of the four-fermion measurement.

**Operation 4: On-patch magic-state preparation**

We inject a noisy magic state into the patch via gate teleportation. First, we prepare the top two qubits in the...
Prep right patch with op. 1.

Fuse patches.

Measure qubits in right patch.

FIG. 9: (Color online.) On-patch pair fusion. To fuse the two fermions indicated in red, bring in a fresh patch in a known state, perform a non-destructive four-fermion measurement between the patches using lattice surgery, and then destructively measure the extra patch by measuring each of its qubits in the $Z$ basis.

(a) $|+\rangle|T\rangle|0\rangle|0\rangle$.
(b) Checks measured.

FIG. 10: (Color online.) On-patch magic-state preparation. (a) One qubit is prepared in $|T\rangle := |T\rangle|+\rangle$, one in $|+\rangle$ and two in $|0\rangle$. (b) Measuring the checks injects the magic state into the fermion pair on the right (or left, up to stabilizer operations.

Operation 5: On-patch exchange

To implement this operation, we can perform the $d = 2$ NISQ version of either the Brown-Roberts [54] or Litinski [29] lattice-surgery implementation of the logical-qubit quantum circuit depicted in Fig. 5. For simplicity, we only describe one of these, the Litinski version. Fig. 11 depicts the steps of this protocol.

In the first step of the Litinski protocol, one first deforms the tetron to be “double wide.” Then one performs a twisted lattice surgery with an ancilla tetron, implementing the $YX$ measurement from Fig. 5. By measuring the qubits in the original patch in the $Z$ basis, one completes this circuit, teleporting the tetron to the new patch with the fermions exchanged as intended (up to Majorana/Pauli frame tracking). If desired, one can perform another $XX$ lattice surgery, followed by measuring the ancilla tetron’s physical qubits in the $Z$ basis, to teleport the fermions back to the original patch with the fermions suitably swapped.

Operation 6: Between-patch exchange

Before describing how to implement a between-patch exchange of Majorana fermions, it is helpful to note that one can shift a tetron laterally one unit, changing the colors (and Pauli types) of the relevant checks. Fig. 12 depicts an example of how to do this. Although the fermion indicated by the black dot in this figure appears to shift in this process, the overall stabilizer constraint for the tetron means that the fermion can equally well be thought of as residing in its original location relative to the now-shifted tetron.

Using this trick, we can compactify the two-tetron to one-hexon fusion operation from Fig. 7 even further to save on qubits for our NISQ realization by using the pro-
FIG. 12: (Color online.) A method to shift a patch a lattice site to the right.

FIG. 13: (Color online.) Between-patch exchange: Part I. The fermions to be exchanged are colored in black and red. The tetrons are fused into a hexon following the prescription of Fig. 7, albeit with a lattice shift on one of the tetrons. On-patch exchange of the relevant fermions via operation 5 is carried out as indicated in Fig. 14, and then all steps are reversed to restore the original two tetrons.

As usual, each of the steps in this protocol involves repeating the measurement of the syndrome twice and comparing them to detect errors.

Once the two fermions are on the same hexon, we can apply operation 5 to the hexon, as depicted in Fig. 14, and then reverse the lattice surgery described previously to restore the original patches.

Operation 7: Between-patch non-destructive four-fermion measurement of neighboring fermions

As described earlier, this operation corresponds to ordinary lattice surgery between two surface code patches. In Fig. 15, the four fermions to be measured are indicated by solid red dots. Because of the stabilizer freedom in both original tetron codes, the red dots may be interpreted as being on the outer vertices instead, so that the measurement of the new check does not destroy them—it merely returns the eigenvalue of the relevant four-fermion observable. When the two patches are split again, the values of the dark half-moon checks in between are either both +1 or both −1, but their correlation belies the fact that they are now entangled by the four-fermion measurement. To be fault-tolerant, both the fusion and splitting steps each are done twice, with any disagreements between the outcomes thrown out.

V. CONCLUSION

Inspired by Feynman’s call to simulate quantum mechanics with quantum computers to eliminate the overhead in quantum-to-classical mappings [1], we have shown how to simulate fermions with error-corrected logical Majorana fermions in a way that eliminates the overhead in fermion-to-qubit mappings. We did so by processing the logical Majorana fermions stored as twist defects in surface code patches. Our approach relies heavily on lattice-surgery-like operations that allow us to mimic measurement-based topological quantum compu-
suggests that our approach is extendable in a way that the simple Majorana fermions we have considered. This anyons as twist defects in topological codes, even beyond that we describe here. We look forward to seeing how the broad array of interest. We expect that similar constructions that utilize braiding of surface-code punctures or transversal operations are also possible, but we also expect that, as is true in the logical qubit case, a puncture-based approach will incur a higher physical-qubit cost and a transversal approach will be more challenging to implement in 2D hardware architectures. We also expect that our constructions will apply to topological codes more broadly than just the surface code, including those defined generally by rotation systems [35] and those defined dynamically, such as the honeycomb code [70, 71].

The ability to manipulate logical Majorana fermions directly in fault-tolerant quantum simulations suggests that one could optimize known quantum simulation algorithms differently. For example, elaborate nonlocal sequences of CNOT gates used to facilitate operations in the Jordan-Wigner mapping of qubits to fermions, such as those described in Refs. [72–75] could be optimized away. We believe that developing optimizing compilers that exploit the availability of native (Majorana) fermionic operations is an interesting avenue for further research. Analyzing resource reductions for specific fermionic quantum simulations is beyond the scope of this work; our object here is to provide tools for others to use to apply to their own quantum simulations of interest. We look forward to seeing how the broad array of fermionic quantum simulation applications will be able to exploit the elimination of fermion-to-qubit mappings that we describe here.

As shown in Ref. [53], one can realize a rich panoply of anyons as twist defects in topological codes, even beyond the simple Majorana fermions we have considered. This suggests that our approach is extendable in a way that facilitates low-overhead FASQ anyonic simulation generally, using these logical anyons in the MBTQC paradigm. For example, the Fradkin-Kadanoff transformation for parafermions that generalizes the Jordan-Wigner transformation for fermions might be able to be eliminated to facilitate less resource intensive studies of parafermions with quantum computers [76, 77]. Simulations of anyonic physics might even help to develop technology based on actual anyonic excitations in material systems [78].

While our approach may facilitate simulation studies of anyonic physics, including of Majorana fermions themselves, the fact that our constructions allow one to manipulate arbitrarily reliable “synthetic” logical Majorana fermions directly suggests that our approach could be an alternative to manipulating Majorana fermions realized as quasiparticle excitations in condensed matter systems for the purposes of reliable quantum computation. Much of the effort developed for how to manipulate Majorana fermions for the purposes of quantum computation, for example protocols for “topological quantum compiling,” [79] can be mapped to the logical Majorana fermion setting without any modifications. That said, the quest to realize physical Majorana fermions is still very important for fundamental physics and could be enabling for some quantum technologies.

It is worth noting that tailoring quantum error correcting codes for explicit use in fermionic quantum simulation algorithms is not a new idea; for example, see Refs. [10–14]. However, all previous constructions of which we are aware either used ad hoc codes or only worked at fixed code distances which did not facilitate arbitrarily reliable quantum simulations. By basing our approach on surface codes, which have been studied extensively, our constructions will work at arbitrary code distances and could be realized by technologies built around surface codes.

While our focus has been on the future beyond the noisy intermediate scale quantum (NISQ) era, we thought it would be helpful to show that progress towards this vision can be established by small-scale NISQ testbed experiments today. To that end, we proposed realizations for $d = 2$ versions of a universal set of fermionic gates, each of which uses only 13 qubits, or 23 if the processing is restricted to be local on a 2D grid. Experimental demonstrations of these and related protocols are important milestones on the path to realizing low-overhead fermionic quantum simulations that offer a true “quantum advantage” over state-of-the-art simulations performed on conventional (classical) computers today.

**Acknowledgments**

This paper benefited from helpful conversations from a number of people, including (in alphabetical order) Andrew Baczewski, Benjamin Brown, Riley Chien, Anand...
Ganti, Markus Kesselring, Daniel Litinski, Jesse Lutz, Setso Motedi, Jamie Stephens, and James Whitfield. We would like to thank the following people for their helpful feedback on an early draft of this paper: Andrew Bačzewski, Lucas Kocia, Kenny Rudinger, Stefan Seritan, and Wayne Witzel. Finally, we thank Daniel Litinski, for inspiring the color scheme we used to depict surface codes.

This material is based upon work supported by the U.S. Department of Energy, Office of Science, National Quantum Information Science Research Centers, the National Nuclear Security Administration’s Advanced Simulation and Computing Program, and the Laboratory Directed Research and Development program at Sandia National Laboratories.

Sandia National Laboratories is a multimission laboratory managed and operated by National Technology and Engineering Solutions of Sandia, LLC., a wholly owned subsidiary of Honeywell International, Inc., for the U.S. Department of Energy’s National Nuclear Security Administration under contract DE-NA-0003525.

This paper describes objective technical results and analysis. Any subjective views or opinions that might be expressed in the paper do not necessarily represent the views of the U.S. Department of Energy or the United States Government.

[1] R. P. Feynman, *Simulating physics with computers*, Int. J. Theo. Phys. 21, 467 (1982), doi:10.1007/BF02650179.
[2] S. Lloyd, *Universal quantum simulators*, Science 273, 1073 (1996), doi:10.1126/science.273.5278.1073.
[3] I. M. Georgescu, S. Ashab, and F. Nori, *Quantum Information Science (QIS) Subcommittee*, Nuclear Science Advisory Committee (NSAC) Quantum Information Science Report (2021), doi:10.1007/BF02650179.
[4] B. Bauer, S. Bravyi, M. Motta, and G. K.-L. Chan, *Report on a National Science Foundation Workshop on enabling quantum leap: Quantum algorithms for quantum chemistry and materials*, in Report on a National Science Foundation Workshop, Alexandrina, VA: January 21–24, 2019 (2019), arXiv:2001.03685, URL https://www.nsf.gov/mps/che/workshops/quantum_algorithms_for_chemistry_and_materials_report_01_21-24_2019.pdf.
[5] Nuclear Science Advisory Committee (NSAC) Quantum Information Science (QIS) Subcommittee, *Nuclear Physics and Quantum Information Science (2019)*, URL https://science.osti.gov/-/media/np/pdf/Reports/NSAC_QIS_Report.pdf.
[6] F. Verstraete and J. I. Cirac, *Mapping local Hamiltonians of fermions to local Hamiltonians of spins*, J. Stat. Mech.: Theory and Experiment 2005, P09012 (2005), doi:10.1088/1742-5468/2005/09/P09012, arXiv:cond-mat/0508353.
[7] J. T. Seeley, J. W. Richard, and P. J. Love, *Optimal fermion-to-qubit mapping via ternary trees with applications to reduced quantum states learning*, Quantum 4, 276 (2020), doi:10.22331/q-2020-06-04-276, arXiv:1910.10746.
[8] Y. Li, *Fault-tolerant fermionic quantum computation based on color code*, Phys. Rev. A 98, 012336 (2018), doi:10.1103/PhysRevA.98.012336, arXiv:1709.06245.
[9] J. von Neumann, *Mathematical Foundations of Quantum Mechanics*, vol. 2 of Investigations in physics (Princeton University Press, Princeton, NJ, 1932 (Reprinted in paperback, 1955)).
[10] Z. Wang, *Topological quantum computation*, in *Topological Quantum Computation* (AMS, 2010), no. 112 in CBMW Regional Conference Series in Mathematics, p. 115.
[11] J. K. Pachos, *Introduction to Topological Quantum Com-
quantum error correction with Majorana zero modes, New J. Phys. 18, 123027 (2016), doi:10.1088/1367-2630/aa50b6, arXiv:1607.07475.

[27] C. Chamberland, P. Iyer, and D. Poulin, Fault-tolerant quantum computing in the Pauli or Clifford frame with slow error diagnostics, 2, 43 (2018), doi:10.22331/q-2018-01-04-43, arXiv:1704.06662.

[28] D. Litinski, A game of surface codes: Large-scale quantum computing with lattice surgery, Quantum 3, 126 (2019), doi:10.22331/q-2019-03-05-126, arXiv:1808.02892.

[29] M. Barkeshli and M. Freedman, Modular transformations through sequences of topological charge projections, Phys. Rev. B 94, 165108 (2016), doi:10.1103/PhysRevB.94.165108, arXiv:1602.01093.

[30] A. Tran, A. Bocharov, B. Bauer, and P. Bonderson, Optimizing Clifford gate generation for measurement-only topological quantum computation with Majorana zero modes, SciPost Phys. 8, 091 (2020), doi:10.21468/SciPostPhys.8.6.091, arXiv:1909.03002.

[31] S. Bravyi, B. Leemhuis, and B. M. Terhal, Majorana fermion codes, New J. Phys. 12, 083039 (2010), doi:10.1088/1367-2630/12/8/083039, arXiv:1004.3791.

[32] D. Gottesman, Stabilizer codes and quantum error correction, Ph.D. thesis, Caltech (1997), arXiv:quant-ph/9705052.

[33] S. Vijay and L. Fu, Quantum error correction for complex and Majorana fermion qubits (2017), arXiv:1703.0459.

[34] R. Sarkar and T. J. Yoder, A graph-based formalism for surface codes and twists (2021), arXiv:2101.09349.

[35] D. Litinski and F. van Oppen, Quantum computing with Majorana fermion codes, Phys. Rev. B 97, 205404 (2018), doi:10.1103/PhysRevB.97.205404, arXiv:1801.08143.

[36] S. Bravyi, Universal quantum computation with the $nu=5/2$ fractional quantum Hall state, Phys. Rev. A 73, 042313 (2006), doi:10.1103/PhysRevA.73.042313, arXiv:quant-ph/0511178.

[37] M. Hastings, T. Karzig, P. Bonderson, M. Freedman, R. Lutchyn, and C. Nayak, Quantum computing methods and devices for Majorana tetron qubits (pending), URL https://patents.google.com/patent/US20180052806A1/en.

[38] A. Y. Kitaev, Quantum error correction with imperfect gates, in Proceedings of the Third International Conference on Quantum Communication, Computing and Measurement, edited by O. Hirota, A. S. Holevo, and C. M. Caves (Plenum Press, New York, 1997), p. 181.

[39] A. Y. Kitaev, Fault-tolerant quantum computation by anyons, Ann. Phys. 303, 2 (2003), doi:10.1016/S0003-4916(02)00018-0, arXiv:quant-ph/9707021.

[40] A. Y. Kitaev, Quantum computations: Algorithms and error correction, Russian Math. Surveys 52, 1191 (1997), doi:10.1070/RM1997v052n06ABEH002155.

[41] S. B. Bravyi and A. Y. Kitaev, Quantum codes on a lattice with boundary (1998), arXiv:quant-ph/9811052.

[42] M. H. Freedman and D. A. Meyer, Projective plane and planar quantum codes, Found. Comput. Math. 1, 325 (2001), doi:10.1007/s102080010013, arXiv:quant-ph/9810055.

[43] H. Bombin and M. A. Martin-Delgado, Topological quantum error correction with optimal encoding rate, Phys. Rev. A 73, 062303 (2006), doi:10.1103/PhysRevA.73.062303, arXiv:quant-ph/0602063.

[44] H. Bombin and M. A. Martin-Delgado, Optimal resources for topological two-dimensional stabilizer codes: Comparative study, Phys. Rev. A 76, 012305 (2007), doi:10.1103/PhysRevA.76.012305, arXiv:quant-ph/0703272.

[45] J. T. Anderson, Homological stabilizer codes (2011), arXiv:1107.3502.

[46] J. P. B. Ataides, D. K. Tuckett, S. D. Bartlett, S. T. Flammia, and B. J. Brown, The XZZX surface code, 12, 2172 (2021), doi:10.1038/s41467-021-22274-1, arXiv:2009.07851.

[47] A. Kitaev and L. Kong, Models for gapped boundaries and domain walls, Commun. Math. Phys. 313, 351 (2012), doi:10.1007/s00220-012-1500-5, arXiv:1104.5047.

[48] Y.-Z. You and X.-G. Wen, Projective non-Abelian statistics of dislocation defects in a $\pi$ rotor model, Phys. Rev. B 86, 161107(R) (2012), doi:10.1103/PhysRevB.86.161107, arXiv:1204.0113.

[49] Y.-Z. You, C.-M. Jian, and X.-G. Wen, Synthetic non-Abelian statistics by Abelian anyon condensation, Phys. Rev. B 87, 045106 (2013), pre-print has a different title., doi:10.1103/PhysRevB.87.045106, arXiv:1208.4109.

[50] T. J. Yoder and I. H. Kim, The surface code with a twist, Quantum 1, 2 (2017), doi:10.22331/q-2017-04-25-2, arXiv:1612.04795.

[51] B. J. Brown, K. Laubscher, M. S. Kesselring, and J. R. Wootton, Poking holes and cutting corners to achieve Clifford gates with the surface code, Phys. Rev. X 7, 021029 (2017), doi:10.1103/PhysRevX.7.021029, arXiv:1609.04673.

[52] M. S. Kesselring, F. Pastawski, J. Eisert, and B. J. Brown, The boundaries and twist defects of the color code and their applications to topological quantum computation, Quantum 2, 101 (2018), doi:10.22331/q-2018-10-19-101, arXiv:1806.02820.

[53] B. J. Brown and S. Roberts, Universal fault-tolerant measurement-based quantum computation, Phys. Rev. Research 2, 033305 (2020), doi:10.1103/PhysRevResearch.2.033305, arXiv:1811.11780.

[54] A. Lavasani and M. Barkeshli, Low overhead Clifford gates from joint measurements in surface, color, and hyperbolic codes, Phys. Rev. A 98, 052319 (2018), doi:10.1103/PhysRevA.98.052319, arXiv:1804.04144.

[55] M. Barkeshli, P. Bonderson, M. Cheng, and Z. Wang,
Symmetry fractionalization, defects, and gauging of topological phases, Phys. Rev. B 100, 115147 (2019), doi:10.1103/PhysRevB.100.115147, arXiv:1410.4540.

[57] G. Zhu, M. Hafezi, and M. Barkeshli, Quantum origami: Transversal gates for universal quantum computation and measurement of topological order, Phys. Rev. Research 2, 013285 (2020), doi:10.1103/PhysRevResearch.2.013285, arXiv:1711.05752.

[58] S. Vijay, T. H. Hsieh, and L. Fu, Majorana fermion surface code for universal quantum computation, Phys. Rev. X 5, 041038 (2015), doi:10.1103/PhysRevX.5.041038, arXiv:1504.01724.

[59] S. Vijay and L. Fu, Physical implementation of a Majorana fermion surface code for fault-tolerant quantum computation, Physica Scripta T168, 014002 (2016), doi:10.1088/0031-8949/T168/1/014002, arXiv:1509.08134.

[60] D. Litinski and F. von Oppen, Braiding by Majorana fermion surface code for universal quantum computation, Phys. Rev. B 96, 205413 (2017), doi:10.1103/PhysRevB.96.205413, arXiv:1708.05012.

[61] P. Aliferis, D. Gottesman, and J. Preskill, Accuracy threshold for postselected quantum computation, Quant. Inf. Comp. 8, 0181 (2008), arXiv:quant-ph/0703264, URL http://www.rintonpress.com/xxqic8/qic-8-34/0181-0244.pdf.

[62] C. Horsman, A. G. Fowler, S. Devitt, and R. van Meter, Surface code quantum computing by lattice surgery, New J. Phys. 14, 123011 (2012), doi:10.1088/1367-2630/14/12/123011, arXiv:1111.4022.

[63] A. J. Landahl and C. Ryan-Anderson, Quantum computing by color-code lattice surgery (2014), arXiv:1407.5103.

[64] D. Litinski and F. von Oppen, Lattice surgery with a twist: Simplifying Clifford gates of surface codes, Quantum 2, 62 (2018), doi:10.22331/q-2018-05-04-62, arXiv:1709.02318.

[65] E. Dennis, A. Kitaev, A. Landahl, and J. Preskill, Topological quantum memory, J. Math. Phys. 43, 4452 (2002), doi:10.1063/1.1499754, arXiv:quant-ph/0110143.

[66] A. G. Fowler, M. Mariantoni, J. M. Martinis, and A. N. Cleland, Surface codes: Towards practical large-scale quantum computation, Phys. Rev. A 86, 032324 (2012), doi:10.1103/PhysRevA.86.032324, arXiv:1208.0928.

[67] J. Łodyga, P. Mazurek, A. Grudka, and M. Horodecki, Simple scheme for encoding and decoding a qubit in unknown state for various topological codes, Sci. Rep. 5, 8975 (2015), doi:10.1038/srep08975, arXiv:1404.2495.

[68] Y. Li and S. C. Benjamin, Hierarchical surface code for network quantum computing with modules of arbitrary size (2015), arXiv:1509.07796.

[69] M. Urbanek, B. Nachman, and W. A. de Jong, Error detection on quantum computers improving the accuracy of chemical calculations, Phys. Rev. A 102, 022427 (2020), doi:10.1103/PhysRevA.102.022427, arXiv:1910.00129.

[70] M. B. Hastings and J. Haah, Dynamically generated logical qubits (2021), arXiv:2107.02194.

[71] C. Gidney, M. Newman, A. Fowler, and M. Broughton, A fault-tolerant honeycomb memory (2021), arXiv:2108.10457.

[72] R. Babbush, C. Gidney, D. W. Berry, N. Wiebe, J. McClean, A. Paler, A. Fowler, and H. Neven, Encoding electronic spectra in quantum circuits with linear T complexity, Phys. Rev. X 8, 041015 (2018), doi:10.1103/PhysRevX.8.041015, arXiv:1805.03662.

[73] V. von Burg, G. H. Low, T. Häner, D. S. Steiger, M. Reiher, M. Roetteler, and M. Troyer, Quantum computing enhanced computational catalysis, 3, 033055 (2021), doi:10.1103/physrevresearch.3.033055, arXiv:2007.14460.

[74] J. Lee, D. W. Berry, C. Gidney, W. Huggins, J. R. McClean, N. Wiebe, and R. Babbush, Even more efficient quantum computations of chemistry through tensor hypercontraction, 2, 030305 (2021), doi:10.1103/PRXQuantum.2.030305, arXiv:2011.03494.

[75] K. Wan, Exponentially faster implementations of Select(H) for fermionic Hamiltonians, p. 380 (2021), doi:10.22331/q-2021-01-12-380, arXiv:2004.04170.

[76] Z.-H. Liu, K. Sun, J. K. Pachos, M. Yang, Y. Meng, Y.-W. Liao, Q. Li, J.-F. Wang, Z.-Y. Luo, Y.-F. He, et al., Topological contextuality and anyonic statistics of photonic-encoded parafermions, PRX Quantum 2, 030323 (2021), doi:10.1103/PRXQuantum.2.030323, arXiv:2011.05008.

[77] E. Fradkin and L. P. Kadanoff, Disorder variables and para-fermions in two-dimensional statistical mechanics, Nucl. Phys. B 170, 1 (1980), doi:https://doi.org/10.1016/0550-3213(80)90472-1.

[78] C. W. J. Beenakker, Search for Majorana fermions in superconductors, Annu. Rev. Con. Mat. Phys. 4, 113 (2013), doi:10.1146/annurev-conmatphys-030212-184337, arXiv:1112.1950.

[79] V. Kliuchnikov, A. Bocharov, and K. M. Svore, Asymptotically optimal topological quantum compiling, Phys. Rev. Lett. 112, 140504 (2014), doi:10.1103/PhysRevLett.112.140504, arXiv:1310.4150.