Optimizing Graph Codes for Measurement-Based Loss Tolerance

Thomas J. Bell[1,2,*] Love A. Pettersson,[3] and Stefano Paesani[3,†]

[1]*Quantum Engineering Technology Labs, H. H. Wills Physics Laboratory and Department of Electrical and Electronic Engineering, University of Bristol, BS8 1FD, United Kingdom*

[2]*Quantum Engineering Centre for Doctoral Training, University of Bristol, United Kingdom*

[3]*Center for Hybrid Quantum Networks (Hy-Q), Niels Bohr Institute, University of Copenhagen, Blegdamsvej 17, Copenhagen DK-2100, Denmark*

Graph codes play an important role in photonic quantum technologies as they provide significant protection against qubit loss, a dominant noise mechanism. Here, we develop methods to analyze and optimize measurement-based tolerance to qubit loss and computational errors for arbitrary graph codes. Using these tools we identify optimized codes with up to 12 qubits and asymptotically large modular constructions. The developed methods enable significant benefits for various photonic quantum technologies, as we illustrate with novel all-photonic quantum repeater states for quantum communication and high-threshold fusion-based schemes for fault-tolerant quantum computing.

Quantum information is fragile and its control can be easily impaired by dissipation in the physical environment. Quantum error correction (QEC) and fault tolerance aim at reducing the impact of noise, provided the physical error rate is below a certain threshold, enabling the control of quantum information in spite of physical imperfections [1–3]. Optimizing codes to the specific platform used and targeting the native noise mechanisms and operations to reduce the operational overheads is key to make QEC practical for near-term and future quantum hardware. In photonics, the dominant noise mechanism is photon loss, which irreversibly erases the state of the associated physical qubit. Although photon loss is an error that can be directly detected, unlike conventional gate errors, it nevertheless poses stringent hardware requirements for practical applications. For example, current architectures for fault-tolerant photonic quantum computing need losses to be below a threshold of approximately 2% [4], very challenging for photonic setups. A possible modular approach to improve these requirements is to encode each computational qubit in a loss-tolerant code, as pictured in Fig. 1. The encoding and decoding are typically measurement-based, i.e., obtained by sequential destructive measurements on part of an entangled resource state to protect the remaining unmeasured components from errors—an approach particularly suitable for photonics [4,7–9]. As we show in this work, codes with moderate size, less than a few tens of qubits, can already provide significant measurement-based suppression of logical errors due to photon loss on encoded qubits. Previous proposals have considered various types of loss-tolerant codes, e.g., tree graph codes [6] [see Fig. 1(a)] and Bacon-Shor codes [10], whose code structures allow the loss tolerance to be readily analyzed. The use of these codes was proposed and investigated, for example, in the context of photonic measurement-based quantum communication [11–14] and computation [4,15]. Identifying resource-efficient codes with high loss tolerance could bring significant practical benefits to these technologies. Here, we address this goal by developing methods to analyze the loss and error tolerance in general graph codes and use them to design and implement optimization techniques. We fully characterize measurement-based fault-tolerant properties and optimize graph codes with up to 12 qubits, and investigate generalizations to larger graphs with modular structures. We find optimized codes that can provide significant advantages in various photonic applications, including improved repeater graph states for quantum communication and fusion-based schemes for fault-tolerant photonic quantum computing with loss thresholds up to 10.5% using standard linear optical fusions.

_____

*tj.bell@bristol.ac.uk
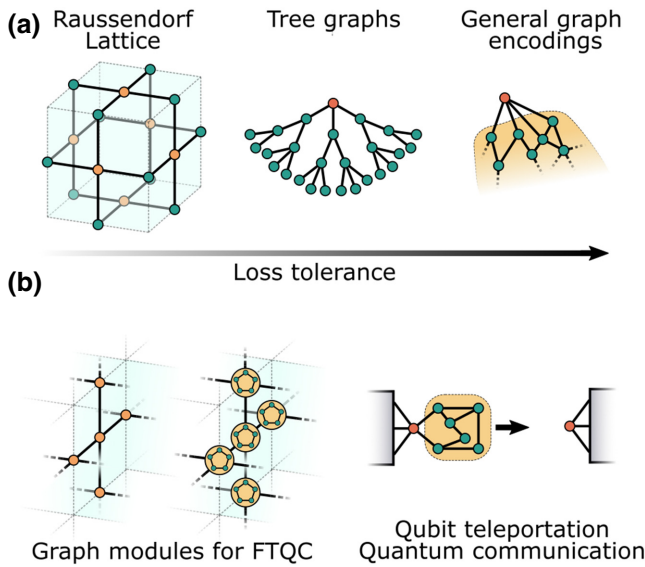
†stefano.paesani@nbi.ku.dk

FIG. 1. (a) Graph codes can encode quantum information with inherent robustness to qubit loss. Example of loss-tolerant graphs include the Raussendorf lattice [5] and tree graphs [6]. In this work we develop tools to analyze the loss tolerance for arbitrary graph codes. (b) Graph codes with improved performance can enable designing better modules for modular photonic applications, including computation or communication schemes.

I. GRAPH CODES

A. Encoding a logical qubit

Graph codes, introduced in Ref. [16,17], are quantum codes with graph states as codewords, a class of quantum states that can be conveniently described in terms of graphs and graph transformations [8,18]. The quantum state associated to an undirected and unweighted graph $G = (V, E)$ with vertices V and edges E is

$$|G\rangle = \prod_{(i,j)\in E} CZ_{i,j} |+\rangle^{\otimes|V|}, \tag{1}$$

where $CZ_{i,j}$ represents a controlled-Z operation between qubits i and j, and is represented by edge between the associated vertices. Graph states are stabilizer states [2] with stabilizer generators $K_i = X_i \prod_{k\in\mathcal{N}_i} Z_k$, where i runs over the graph nodes and $\mathcal{N}_i$ is the neighborhood of qubit i. Throughout this work, we use X, Y, Z to indicate Pauli operators. The generators generate the Abelian group of stabilizer operators $\mathcal{S} = \langle K_j \rangle_{j=1}^n$, meaning that each stabilizer $S \in \mathcal{S}$ is a product of generators $S = \prod_{i=1}^n K_i^{b_i}$, with $b_i \in \{0, 1\}$.

A possible method to encode an arbitrary qubit state $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$ in an n-qubit graph code is pictured in Fig. 2(a). We initially prepare the n physical code qubits in a graph state $|G\rangle$ and an additional input qubit in the target state $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$. The logical encoding is then performed by applying controlled-Z operations between

the input qubit and a subset B of the code qubits, and then measuring the input qubit in the X basis. As these operations are all in the Clifford group, we can track the stabilizers and logical operations on the state according to the usual stabilizer transformation rules [2], obtaining the n-qubit code in the logical state $|\overline{\psi}\rangle = \alpha|\overline{0}\rangle + \beta|\overline{1}\rangle$ if the measurement outcome is $+1$, and $\overline{Z}|\overline{\psi}\rangle$ on outcome -1. The bar will be used throughout to denote logical elements. Here $|\overline{0}\rangle = |G\rangle$ and the logical operators are given by $\overline{X} = \prod_{b\in B} Z_b$ and $\overline{Z} = K_{b_0}$ for some choice of $b_0 \in B$ [see Fig. 2(c)]. The updated code stabilizers are $\mathcal{S} = \langle K_{b_0} K_b, K_{b'}\rangle_{b\in B\setminus b_0, b'\notin B}$, with K_i the stabilizer generators of $|G\rangle$ [8]. As products of a logical operator with stabilizers also form valid logical operators of the code, we can write the set of all logical operators as $\overline{\mathcal{L}} = (\overline{X}, \overline{Z}, \overline{Y} = i\overline{X}\,\overline{Z}) \cdot \mathcal{S}$. Note that an equivalent approach to encoding can be obtained by initializing the input qubit in $|+\rangle$ before applying the controlled-Z operations to the code qubits in B, and then measuring it in the qubit basis $\{|\psi^*\rangle, |\psi_\perp^*\rangle\}$ (which in general can be non-Clifford), as pictured in Fig. 2(b). Here we define $|\psi^*\rangle = \alpha^*|0\rangle + \beta^*|1\rangle$ to be the complex conjugate of $|\psi\rangle$. In this way, the encoding can be described as starting from a fixed *progenitor* graph $G' = (V + \{\text{input}\}, E + (\{\text{input}\}, B))$ with $n + 1$ nodes, and then measuring the input node in the targeted basis. This can be simply observed by writing the initial graph state as $|G'\rangle = (|0\rangle_{\text{in}}|G\rangle + |1\rangle_{\text{in}} \prod_{b\in B} Z_b|G\rangle) / \sqrt{2} = (|0\rangle_{\text{in}}|\overline{0}\rangle + |1\rangle_{\text{in}}|\overline{1}\rangle) / \sqrt{2}$ with codewords and logical operators defined as above, for which projective measurement of the input qubit into $|\psi^*\rangle$ prepares the logical code state in the conjugate state $|\overline{\psi}\rangle$, as desired. The two pictures are equivalent but, depending on the protocol under study, analyzing the encoding in the progenitor picture may be more convenient as it permits description of the encoding and decoding of a logical qubit entirely through the stabilizers of the progenitor graph. In fact, the stabilizer generators of the progenitor graph that act as X_{in} and Z_{in} on the input qubit are transformed in the logical operators $\overline{X}$ and $\overline{Z}$ of the code, respectively, upon measurement of the input. That is, $\overline{X} \leftrightarrow K'_{\text{in}}$ and $\overline{Z} \leftrightarrow K'_{b_0}$, where K'_i are the stabilizer generators of the progenitor graph G' and $b_0 \in B$. Note that the encoding methods discussed above rely on physical operations on the input qubit, and could thus be noisy, as loss or error in the input physical qubit may correspond to logical errors. Therefore, while convenient analysis tools and potentially noise tolerant in specific implementations, they are not general prescriptions for fault-tolerant state preparation.

B. Measurement-based decoding

Measurement-based approaches process logical quantum information by only performing destructive single-qubit measurements on the code qubits and classical feed forward. In the context of QEC, the constraints imposed by
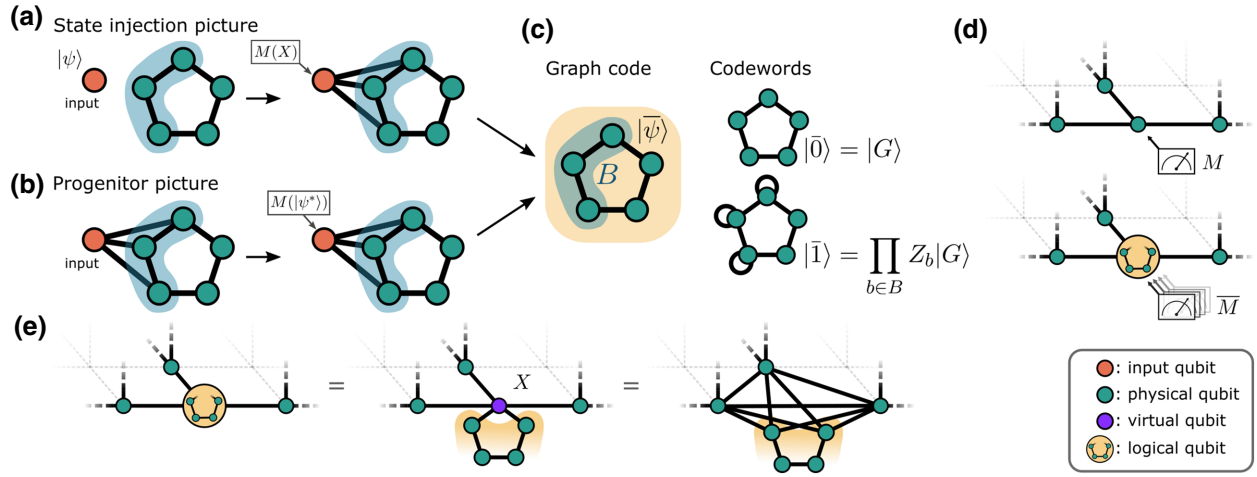
FIG. 2. Encoding in graph states. (a) State injection picture, where the logical state is encoded by preparing it on an ancilla qubit, followed by entangling operations and an X measurement on the ancilla. (b) Progenitor picture, where the same encoding is performed measuring the input qubit of the progenitor graph in the targeted state. (c) The prepared graph code in the logical state $|\overline{\psi}\rangle$, and associated codewords. Loop edges represent Z Pauli operations on the associated qubits [19]. (d) Modular schemes for measurement-based operations, where computational qubits are encoded in graph codes and computational qubit measurements correspond to logical measurements on the graph codes. (e) Physical modularization of resource states, obtained substituting each computational qubits with a virtual qubit, the input of the associated graph code, to be measured in X.

operating a single destructive measurement per qubit add significant limitations with respect to repeatedly performing parity checks in circuit-based approaches. Namely, given a single-qubit measurement pattern M on the physical qubits, the only stabilizers that will be accessible are those that commute qubitwise with M, i.e., the stabilizers in

$$\mathcal{S}_M = \{S \in \mathcal{S} \mid [S_i, M_i] = 0 \text{ for each qubit } i\}, \quad (2)$$

with $\mathcal{S}$ the initial code stabilizers. Note that $\mathcal{S}_M \subseteq \mathcal{S}$ forms a stabilizer subgroup of $\mathcal{S}$ (Appendix E). Therefore, the effect of these constraints is effectively to induce a reduced code $\mathcal{S}_M$ compatible qubitwise with the measurement M. This can also be regarded, more abstractly, as a gauge-fixing procedure [20,21]. Measurement-based decoding of gate errors, i.e., the inference of qubit errors from the measured syndromes, can then be performed equivalently as one would do in standard QEC by considering the reduced code $\mathcal{S}_M$ induced by M.

C. Effects of qubit loss

The effect of qubit loss detected during measurements can be described similarly to the enforcement of a measurement pattern M described above. If a qubit is lost, all stabilizers and logical operators that act nontrivially on that qubit are no longer measurable. This enforces a qubitwise constraint similar but stronger to Eq. (2) as compatibility now requires an identity on a lost qubit rather than just a commuting operator. To maintain a concise notation when describing the effects of loss, we write $M_i = \not\exists$ to indicate

that qubit i was lost, and use the convention that $[A, \not\exists] = 0$ iff $A = \mathbb{1}$. With this notation, we can write the set of stabilizers $\mathcal{S}_M$ compatible with M, which now includes also lost qubits, again exactly as Eq. (2). Also in this case $\mathcal{S}_M$ is a stabilizer subgroup of the initial stabilizer group $\mathcal{S}$; the presence of losses has the effect of reducing it further (see Appendix E for more details). However, qubit losses also pose constraints on logical operators of the code as they cannot have support on a lost qubit. These constraints can be included in a very similar way as for the stabilizers by writing the induced set of logical operators as

$$\overline{\mathcal{L}}_M = \{\overline{L} \in \overline{\mathcal{L}} \mid [\overline{L}_i, M_i] = 0 \text{ for each qubit } i\}, \quad (3)$$

again using the convention $M_i = \not\exists$ if qubit i is lost. In the progenitor graph picture, the conditions in Eq. (3) can be conveniently included by directly applying Eq. (2) to the stabilizers of the progenitor graph. The main idea behind loss-tolerant measurement-based QEC is that, if losses are not excessive, the set $\overline{\mathcal{L}}_M$ remains nontrivial and $\mathcal{S}_M$ contains enough stabilizers to protect the encoded logical state from errors. In general, the code performance depends on the chosen single-qubit measurement pattern M, as well as on the initial graph code. If losses are *heralded*, i.e., which qubits are lost is known before their measurement, the measurement pattern M can be conveniently optimized beforehand to achieve the best available $\mathcal{S}_M$ and $\overline{\mathcal{L}}_M$ [22]. However, losses are often *unheralded*: loss of a qubit is detected only upon its measurement and not before. This loss model is relevant to most quantum platforms (e.g., photonics), and we focus on it in this work. Finding initial graph codes and measurement strategies that provide

good tolerance to unheralded loss is in general a complex task and is investigated in the next sections.

II. LOSS-TOLERANT LOGICAL MEASUREMENTS WITH GRAPH CODES

In measurement-based approaches, computational operations are implemented via single-qubit measurements. We start describing general methods to perform these measurements loss tolerantly when encoding each computational qubit into an arbitrary graph code, as depicted in Figs. 2(d)–2(g).

A. Loss-tolerant logical Pauli measurements

As stabilizers are based on the Pauli group, logical measurements in the Pauli bases are the simplest to analyze within the framework we describe: it corresponds to measuring a logical operator $\overline{L} \in \{\overline{X}, \overline{Y}, \overline{Z}\}$. In the progenitor picture, this can be seen as a nondestructive Pauli measurement on the input qubit without having to directly measure it, often referred to as an *indirect* measurement [6]. Let us consider, for example, an indirect measurement of the $\overline{X}$ logical operator. In the presence of qubit loss, a measurement pattern M on the physical qubits successfully measures it if the set $\overline{\mathcal{L}}[X]_M$, obtained applying the condition in Eq. (3) to the set $\overline{\mathcal{L}}[X]$ of all possible logical $\overline{X}$ operators of the code, is nonempty. Equivalently, it means there exists a logical operator $\overline{X} \in \overline{\mathcal{L}}[X]_M$ that can be obtained from the single-qubit measurements performed in M and with no support on lost qubits (i.e., $\overline{X}_i = \mathbb{1}$ if $M_i = \not\exists$). Identical conditions apply for $\overline{Y}$ and $\overline{Z}$ indirect measurements. A simple example is the indirect logical $\overline{X}$ measurement on star-graph codes, i.e., graph codes with a star-graph progenitor [see Fig. 3(a)]. From the definitions in Sec. I, it is evident that a single-qubit operator X_i on any code qubit i provides a valid logical $\overline{Z}$ operator, i.e., $\overline{\mathcal{L}}[Z] = \{X_i\}_{i=1}^n$. Therefore, choosing the measurement pattern $M = \prod_i X_i$, if at least one code qubit is not lost then $\overline{\mathcal{L}}[Z]_M$ is nonempty and the logical operator is successfully measured. The probability of failing to measure $\overline{Z}$, which we call logical loss, is thus $\overline{\ell}[Z] = \ell^n$ where n is the number of physical code qubits and ℓ the qubit loss probability. Such strong robustness for $\overline{Z}$ measurements for the star graphs comes at the expense of weak performance for $\overline{X}$ and $\overline{Y}$ measurements. In fact, both $\overline{X}$ and $\overline{Y}$ are weight-n operators, meaning that all physical qubits need to exist to obtain a successful logical measurement, so $\overline{\ell}[X] = \overline{\ell}[Y] = 1 - (1 - \ell)^n$.

In arbitrary graph codes, the optimality of a measurement pattern M on the remaining undetected qubits may depend on the losses detected on already-measured qubits. Therefore, given a graph code, an important task is now finding a decoding strategy that optimizes the probability to achieve a measurement pattern M providing a successful logical measurement. Specifically, while decoding in
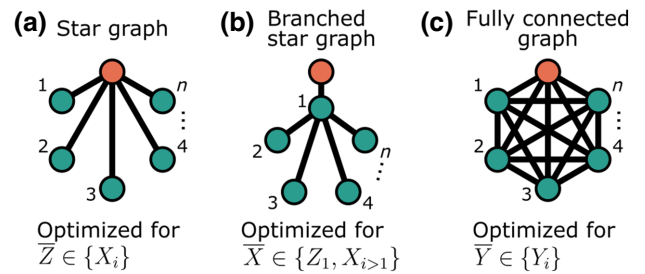

FIG. 3. Optimal graphs for individual logical Pauli measurements. (a) Star graph providing optimal loss tolerance for measuring $\overline{Z}$, (b) branched star graph for $\overline{X}$, and (c) star graph for $\overline{Y}$. Considering n code qubits, for any of these graph structures the success probability to measure the associated logical operator is $1 - \ell^n$, although for the remaining two logical Pauli operators it decreases exponentially as $(1 - \ell)^n$, with ℓ the physical loss rate.

the presence of unheralded loss we need to consider: (1) a single-qubit measurement pattern M consistent with the measurements already performed, (2) a rule that determines which of the unmeasured qubits should be measured next, (3) a rule that allows us to update the decoding strategy as new qubit loss is detected. The general structure for the algorithm we consider to optimize loss-tolerant decoding strategies is described in Algorithm 1. It is an iterative algorithm where at each iteration a new measurement is decided via a NextMeas function, and the available operators $\overline{\mathcal{L}}[A]$ updated via UpdateDecoder, which implements the constraints in Eq. (3). For moderate-size codes we can analyze the decoding procedure in terms of a decision tree describing the evolution of the decoder status conditional on the qubit measurements. The resource-intensive process of building the decision tree can be done offline prior to execution, such that runtime costs are reduced to up to n queries of a look-up table, each possibly followed by adjustment of measurement bases. Such structures also provide us an analytical formula for

---

Input: Set $\overline{\mathcal{L}}[A]$ of logical $A \in \{X, Y, Z\}$ code operators.
Output: Outcome of $\overline{A}$ if successful, False otherwise.

1 Initialize the set of unmeasured qubit $\Theta = \{1, \ldots, n\}$ using all n-code qubits, and $M = \mathbb{1}^{\otimes n}$;
2 **while** Θ *and* $\overline{\mathcal{L}}[A]$ *are non-empty* **do**
3     $i, P_i \leftarrow$ NextMeas$(\Theta, \overline{\mathcal{L}}[A])$. ;   ▷ Decide next qubit i and Pauli operator P_i to measure.
4   Measure qubit i, set $M_i = P_i$ if successful, $M_i = \not\exists$ if qubit is lost;
5     $\overline{\mathcal{L}}[A] \leftarrow$ UpdateDecoder$(M, \overline{\mathcal{L}}[A])$; ▷ Update $\overline{\mathcal{L}}[A]$ according to Eq. (3)
6   Remove qubit i from Θ.
7 If $\overline{\mathcal{L}}[A]$ is nonempty return value of $\overline{A} \in \overline{\mathcal{L}}[A]$, else return False.

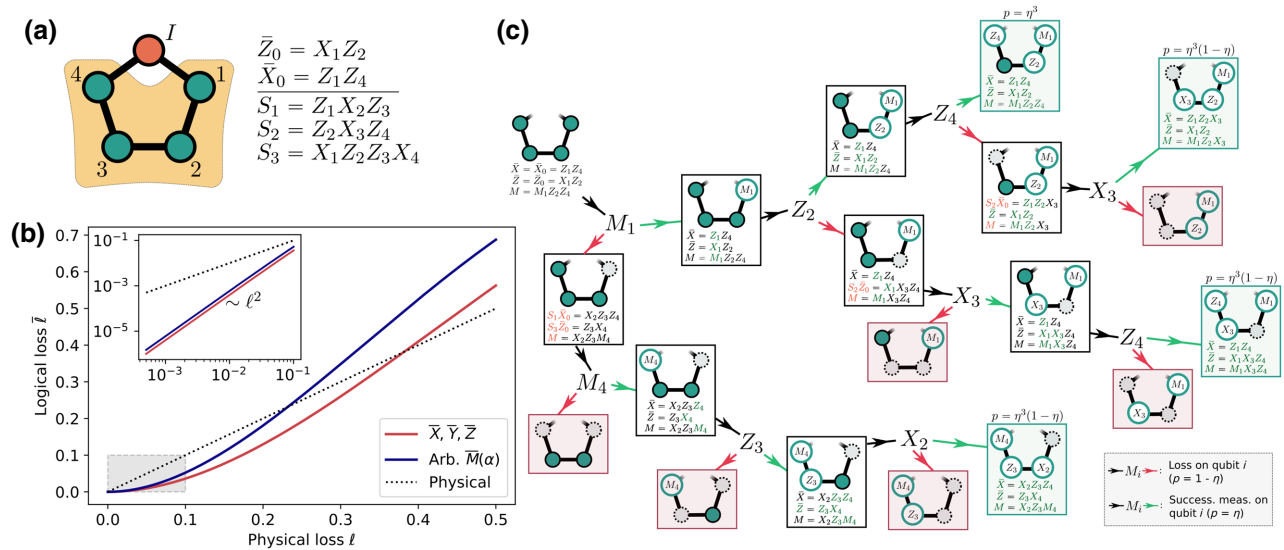ALGORITHM 1. Pauli measurement decoder.

FIG. 4. Loss-tolerant logical measurements with the pentagon graph. (a) Progenitor graph for the pentagon code, with its two logical operators and the list of stabilizer generators. (b) Performance of the pentagon code for all logical Pauli measurements (red) and logical measurements in an arbitrary basis (blue). Physical loss is shown as a dotted black line, with break-even points obtained at 38% and 23% for Pauli and arbitrary measurements, respectively. (c) Exemplary decision tree for decoding an arbitrary measurement on the pentagon graph tolerating the loss of any single qubit. Green arrows represent successful qubit measurements, red arrows represent loss of the measured qubits. At each measurement, the number of valid measurement strategies remaining is reduced.

the logical success probability in terms of the qubit loss ℓ by summing the conditional probabilities of all paths in the tree that end in a successful logical measurement, as exemplified in Fig. 4(c) for the pentagon code. More details on the algorithm and decoding procedures are described in Appendix A.

When testing the graphs in Fig. 3 we retrieve the expected loss tolerance with scaling $\bar{\ell} = \ell^n$, optimal for a single logical Pauli measurement. A more interesting problem is instead to find codes with good loss tolerance for any logical Pauli $\bar{A} \in \{\bar{X}, \bar{Y}, \bar{Z}\}$ measurement, and as such we consider as our metric the worst-case logical loss rate, $\bar{\ell}[\text{Pauli}] = \min\{\bar{\ell}[X], \bar{\ell}[Y], \bar{\ell}[Z]\}$. The fact that loss tolerance is invariant in locally equivalent graphs [18,23] (see Appendix D) allows us to make the optimization procedure more efficient, as we have to only analyze one representative graph state per local-equivalence class for a comprehensive analysis. Detailed categorization of graph-state equivalence classes have been performed for graphs with up to 12 qubits [24,25], which we use to carry out an exhaustive search of small-scale (up to $n = 11$ code qubits, i.e., progenitor graphs with 12 qubits) graph codes. We find that 12-qubit progenitor graph states can be analyzed typically in a few seconds on a standard laptop, but due to the large number of equivalence classes ($>10^6$) we take advantage of the high-performance computing cluster BlueCrystal at the University of Bristol. We identify the pentagon code, shown in Fig. 4(a), as the smallest code ($n = 4$) showing loss tolerance simultaneously for more

than one logical Pauli measurement. For this graph we obtain the same logical success probability $\bar{\eta} = 2\eta^2 - \eta^4$ for logical measurements of any Pauli operator, plotted in Fig. 4(b), where $\eta = 1 - \ell$ is the physical transmittivity and $\bar{\eta} = 1 - \bar{\ell}$. When the physical loss is below $\ell^* \simeq 38\%$ loss tolerance begins to appear as the logical loss is lower than the physical one. This value is denoted the *break-even point*, where the graph encoding outperforms the bare physical qubit, and is often similar but not necessarily equal to the code's loss *threshold* under concatenation, see Sec. IV. At low-loss rates $\bar{\ell} \sim 4\ell^2$ [see Fig. 4(b) inset], indicating the code is able to protect against the loss of any single qubit. We show in Figs. 5(a)–5(c) the results of the optimization for progenitor graphs with up to 12 qubits. In Fig. 5(a) we report some of the graph states (see Appendix F for a complete list) we identify with optimized loss tolerance at loss values much smaller than the break-even point, where we are well within the subthreshold regime (i.e., optimizing at a physical loss level $\ell = 1\%$). The associated performances are plotted in Fig. 5(b). Break-even points up to 50% can already be achieved for these graph codes of moderate size. In some cases, further improvements can also be obtained considering graphs optimized at loss levels close to the break-even point (i.e., $\ell \simeq 30\%$, depending on the code size), represented by the dashed lines in Fig. 5(c), with the associated graph states reported in Appendix F. These two parameter regimes are found to well represent code performance and any improvements from optimizing at different loss values are found to be

small. The tools presented here allow identification of optimized graphs tailored to the noise rates of the physical system.

B. Logical measurements in an arbitrary basis

In the measurement-based framework, Pauli measurements correspond to Clifford operations. In order to perform universal quantum computations, we also require measurements in an arbitrary basis on the Bloch-sphere equator $A(\theta) = X\cos(\theta) + Y\sin(\theta)$ providing non-Clifford operations [26]. In order to perform this operation loss tolerantly on a graph code, the idea we consider can be regarded as adaptive teleportation of the encoded state into a single code qubit preemptively measured in $A(\theta)$. If we know from the start that a physical qubit is not lost, which we call the *output* qubit, then a sufficient condition for measuring $\overline{A}(\theta)$ on $|\overline{\psi}\rangle$ is to measure the other code qubits such that the encoded state is teleported onto the output qubit [22]. However, we can also think of inverting the order of these operations: first attempt to measure $A(\theta)_{out}$ and then do the teleportation only if the output photon is successfully detected, and otherwise try again with a different output qubit [6]. The two orderings provide the same outcome up to a feed-forward operation on the output. Care is hence required to account for the Pauli frame update imposed by the outcome of intermediate measurements. This may be done with classical postprocessing of measurement outcomes in some cases, in other cases more correction circuits may need to be employed, for example, those discussed in Refs. [6,27]. Procedures to perform arbitrary logical measurements can thus be obtained by adapting techniques for loss-tolerant teleportation in graph states to the case where the output is not a fixed qubit. In particular, Ref. [22] provides a sufficient condition for a measurement pattern M to teleport the encoded state to a fixed output qubit, called the *stabilizer pathfinding conditions* (SPCs). In terms of the constraints in Eq. (3), SPCs can be stated as follows: a measurement pattern M of local Pauli operators on code qubits not including the output ($M_{out} = \mathbb{1}$) teleports the encoded state to the output qubit, up to a random but known local unitary $U_{out} \in \{\mathbb{1}, X, Y, Z\}$, if $\overline{\mathcal{L}}_M$ contains two anticommuting logical operators. For logical measurements, to this condition we need to add the successful initial measurement of the output in $A(\theta)_{out}$. The SPC can be included in the decoder very similarly as in the Pauli measurement decoder discussed in the previous section (Algorithm 1). The main differences are as follows: (1) we now need to consider the set of all logical operators $\overline{\mathcal{L}}$, instead of a single logical Pauli, and (2) we require it to contain at least two anticommuting operators at the end, instead of just being nonempty. The approach is thus modified to a decoder structure as described in Algorithm 2, with more details reported in Appendix A.

Input: Set $\overline{\mathcal{L}}$ of all logical operators, operator A.
Output: Outcome of arbitrary logical measurement $\overline{A}$ if successful, `False` otherwise.

1 Initialise $\Theta = \{1, \ldots, n\}$, $M = \mathbb{1}^{\otimes n}$, $outdet = $ `False`;
2 **while** Θ *and* $\overline{\mathcal{L}}$ *are non-empty* **do**
    /* Measure an output qubit   */
3   **while** *!outdet* **do**
4         $out \leftarrow \texttt{NextOut}(\Theta, \overline{\mathcal{L}})$; ▷ Decide next output qubit out to attempt measuring in A_{out}.
5   Attempt A_{out} on qubit out, set $outdet = $ `True` if successful, $M_{out} = \not\exists$ if qubit is lost;
6         $\overline{\mathcal{L}} \leftarrow \texttt{UpdateDecoder}(M, \overline{\mathcal{L}}, out)$;   ▷ Update $\overline{\mathcal{L}}$ according to Eq. 3.
7   Remove qubit i from Θ.
    /* Teleport to the output   */
8     $i, P_i \leftarrow \texttt{NextMeas}(\Theta, \overline{\mathcal{L}}, out)$. ;   ▷ Decide next qubit i and Pauli operator P_i to measure.
9   Measure qubit i, set $M_i = P_i$ if successful, $M_i = \not\exists$ if qubit is lost;
10     $\overline{\mathcal{L}} \leftarrow \texttt{UpdateDecoder}(M, \overline{\mathcal{L}}, out)$;   ▷ Update $\overline{\mathcal{L}}$ according to Eq. 3. Set *outdet* = False if current output no longer feasible.
11   Remove qubit i from Θ.
12 If *outdet* is `True` and $\overline{\mathcal{L}}$ contains a pair of anticommuting operators return value of A_{out} and U_{out} from the SPC, else return `False`.

Algorithm 2.   Arbitrary measurement decoder

The loss tolerance of graph codes under arbitrary basis measurements is again preserved between locally equivalent graphs (see Appendix D), allowing for a streamlined optimization procedure. The smallest code displaying loss tolerance we identify is again the pentagon progenitor graph in Fig. 4(a). The logical success probability for arbitrary $\overline{A}$ measurement with this graph is $\overline{\eta} = 4\eta^3 - 3\eta^4$ [see Fig. 4(b)]. Loss tolerance is observed below a physical loss breakeven point of $\ell^* \approx 23\%$, and we observe a subthreshold scaling of the logical loss as approximately $6\ell^2$ indicating tolerance against the loss of any single code qubit. It can be noted, also comparing the behaviors in Fig. 4(b), that the loss-tolerance performance is worse than logical measurements of Pauli operators, as expected. In Fig. 4(d) we report, for various code sizes, the graphs we identify, which optimize the loss tolerance for arbitrary logical measurements in the subthreshold regime. Their logical loss behavior is shown in Fig. 4(e). We again observe improved loss tolerance for larger codes, and break-even points that are higher by a few percent when optimizing in this regime, as shown in Fig. 4(f). For the largest size explored, $n = 11$ code qubits (12-qubit progenitor graphs), we obtain a break-even point of $\ell^* \simeq 40\%$.
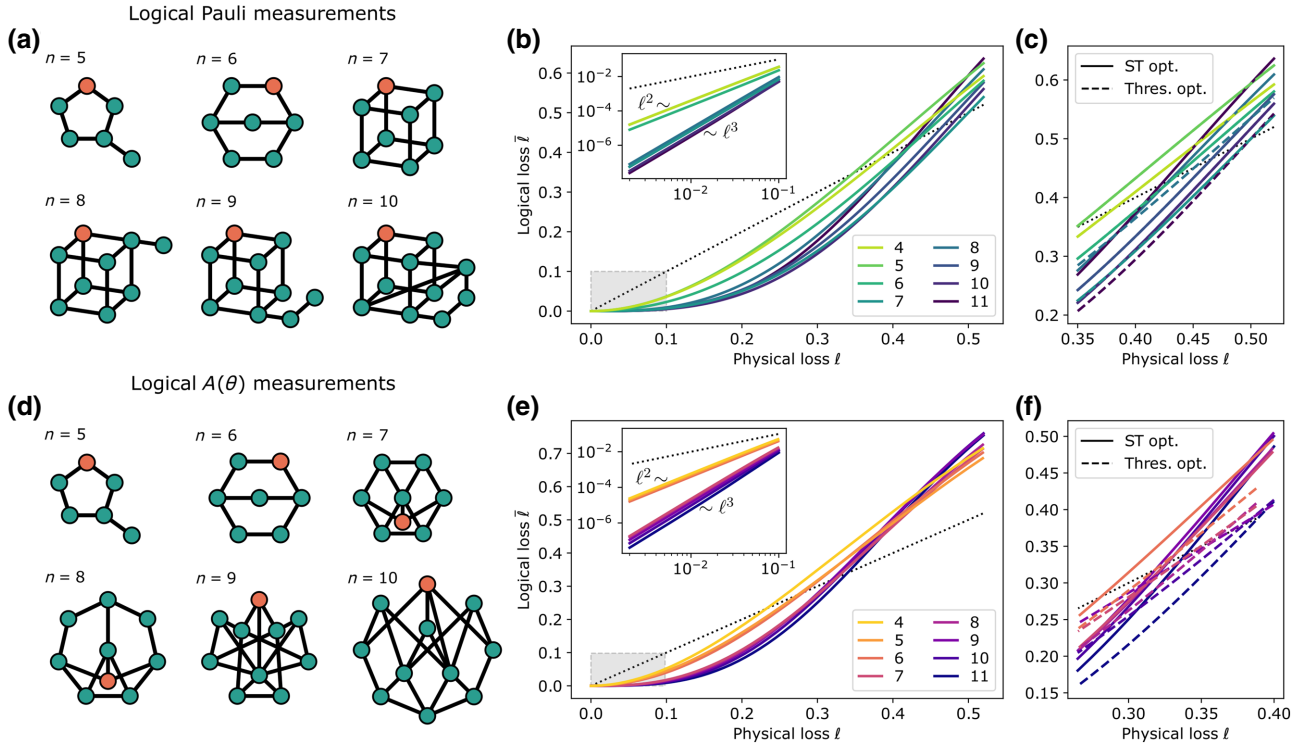
FIG. 5. Optimized graph codes for loss-tolerant measurements. (a) Progenitor graphs of the graph codes optimized for Pauli-basis measurements in the subthreshold regime for different code sizes, with the associated performance shown in (b). Each graph is a representative with minimum edge number in its local-equivalence class. (c) Comparison of near-break-even performance for graphs optimized for their loss break-even point (dashed lines) and the subthreshold-optimized graphs (solid lines). (d)–(f) Analogous plots for arbitrary basis measurements.

III. MEASUREMENT-BASED ERROR CORRECTION IN LOSS-TOLERANT GRAPHS

Qubit errors arising from imperfect gates and measurements can also be simultaneously corrected with graph codes. Unlike photon loss, they cannot be directly detected, so need to be inferred using the code stabilizers $\mathcal{S}_M$ and operators $\overline{\mathcal{L}}_M$ induced by the measurement pattern M, as described in Sec. I.

For logical Pauli measurements, measurement-based error correction corresponds to updating the Pauli frame, meaning that the correction can be done by postprocessing the logical measurement outcome. Specifically, after inferring an error E from a decoder, the outcome of $\overline{L}$ is flipped if the supports of E and $\overline{L}$ share an odd number of qubits (if no $\overline{L}$ exists due to losses, we consider it a logical error as well). For arbitrary logical measurements, the situation is similar but considers both logical operators $(\overline{L}, \overline{K})$ in a pair satisfying the SPCs, as described in Sec. II. Note, however, that in this case we cannot identify and correct errors on the physical $A_{\text{out}}(\theta)$ measurement of the output qubit, as that operation is outside the stabilized space (unless $A_{\text{out}}(\theta)$ is a Pauli measurement). Therefore, the logical error rate for arbitrary measurements cannot be smaller than the physical rate on the output qubit. This is related to the fact

that in the measurement-based framework logical arbitrary measurements $A_{\text{out}}(\theta)$ correspond to arbitrary non-Clifford single-qubit operations, and stabilizer codes possess only a limited set of natively fault-tolerant gates [28,29].

To maintain generality for arbitrary graph codes, we implement error decoding via maximum likelihood, which is computationally viable for the moderate-size codes considered here, and consider a phenomenological error model of independent identically distributed (IID) Pauli errors on each qubit, corresponding to the depolarizing channel $\rho \rightarrow (1 - 3\lambda)\rho + \lambda(X\rho X + Y\rho Y + Z\rho Z)$. More details on the decoding procedure can be found in Appendix A. The error correction is again invariant for locally equivalent graphs, up to permutations of the Pauli bases, which arise from local complementations on the progenitor graph, which effectively act as Clifford operations on the logical operators (see Appendix D). We can thus analyze individual graphs from local-equivalence classes to characterize codes of increasing size, now using models where errors and loss are simultaneously present. Considering indirect Pauli measurements, the smallest progenitor graph found to exhibit fault tolerance against both errors and losses is the cube graph, shown in Fig. 6(a), which generates a code locally equivalent to the seven-qubit

Steane code [30]. As shown in Fig. 6(b), when noises are individually present, it outperforms the bare qubit for losses below 50%, saturating the bound of the measurement complementarity principle [31], and for physical errors $\lambda \leq 3.2\%$. In Fig. 6(c) we plot the overall fault probability in the presence of loss and Pauli errors, where the fault probability is the probability of at least one error type occurring during measurement. In Fig. 6(d) we show the ratio between the logical and physical fault probabilities, where now a break-even curve can be observed and shows a remarkable robustness for this code.

As mentioned above, it is not possible to reduce logical errors below the single-qubit level for arbitrary basis measurements. Still, we find examples of graphs that saturate this linear bound at low error rates. For example, we show in Fig. 6(e) the decorated pentagon graph, which is loss tolerant for arbitrary measurements with a break-even point of 32% [see Fig. 6(f)] while simultaneously having logical error rates $\overline{\epsilon}/\epsilon \to 1$ as $\epsilon \to 0$.

IV. EXTENDING TO LARGER GRAPHS BY MODULARIZATION

To analyze the performance of larger codes efficiently, we consider two modular approaches: cascading and concatenating small *unit graphs*, which can be fully characterized using the previously described techniques.

A. Cascaded graphs

We define cascaded graphs as layered graphs constructed by recursively appending unit graphs to code qubits. In particular, by embedding each code qubit in the kth layer with a unit graph to which it is the input qubit, then the code qubits of all the added unit graphs represent the qubits in the $(k + 1)$th layer. This step can then be repeated to recursively build larger graphs in a modular approach, where the unit graphs used can also vary at different layers. Examples of cascaded graphs are shown in Figs. 7(b) and 7(g). The inspiration for the cascaded graphs' construction comes from tree-graph codes [6], which can be seen as cascades of star graphs [see Fig. 3(a)], locally equivalent to Greenberger-Horne-Zeilinger (GHZ) states. Cascaded graphs are a generalization of tree-graph structures using arbitrary graphs as modules.

The structure of cascaded graphs is such that the analysis of their loss and error tolerance properties can be efficiently obtained once the performance of the small-size unit graphs is known. The idea is to consider the measurement patterns suggested on the top-layer graph, and modify them recursively when going to deeper layers. In fact, the cascaded structure allows us to leverage indirect measurements of qubits in upper layers via the measurement of qubits in deeper layers. For example, if a top-layer qubit is to be measured in the Z basis, it can equally be measured indirectly using qubits restricted to the second layer of the graph, so loss of the qubit can be tolerated. In fact, for any lost qubit at depth k of a cascaded structure, one can attempt to recover an indirect Z measurement by measuring depth $k + 1$ qubits. Measurements in non-Z bases, however, necessarily require additional indirect measurements of deeper qubits, as logical non-Z operators in the progenitor graph have always support on the code qubits
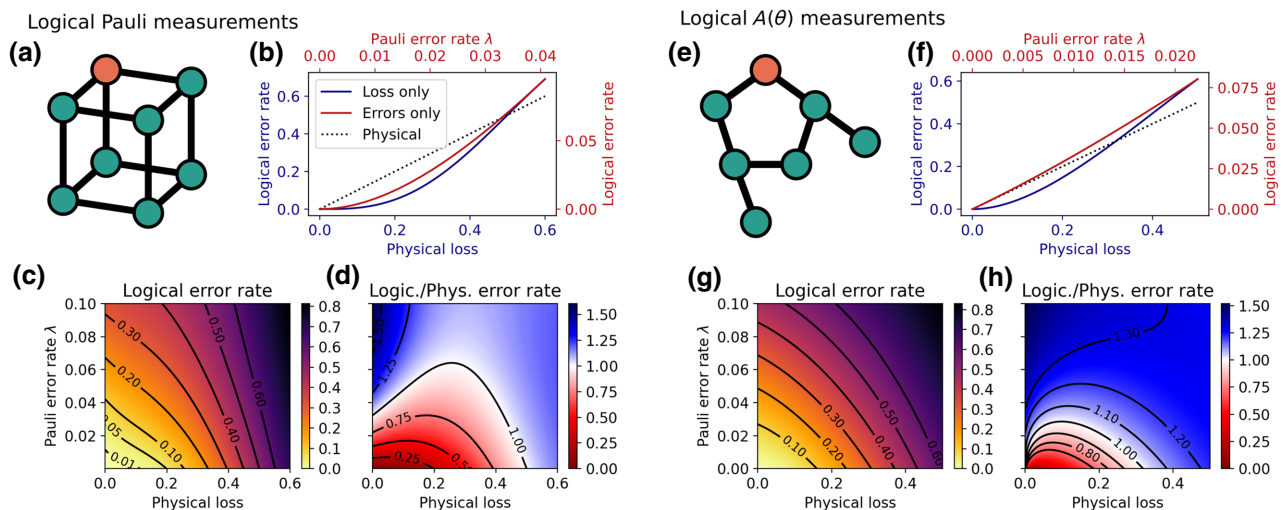

FIG. 6.   Simultaneous correction of loss and errors on graph codes. (a) The cube graph is the smallest progenitor graph able to perform logical measurements in all Pauli bases correcting simultaneously losses and errors. (b) Logical error probability for each noise type in isolation, and (c) when both are simultaneously present. (d) Ratio of encoded to bare error rates, highlighting the region (in red) where error suppression occurs. (e)–(h) Analogous plots for arbitrary basis measurements. The smallest graph able to correct losses and to saturate the scaling $\overline{\epsilon}/\epsilon \to 1$ at low error rates is the decorated pentagon graph.
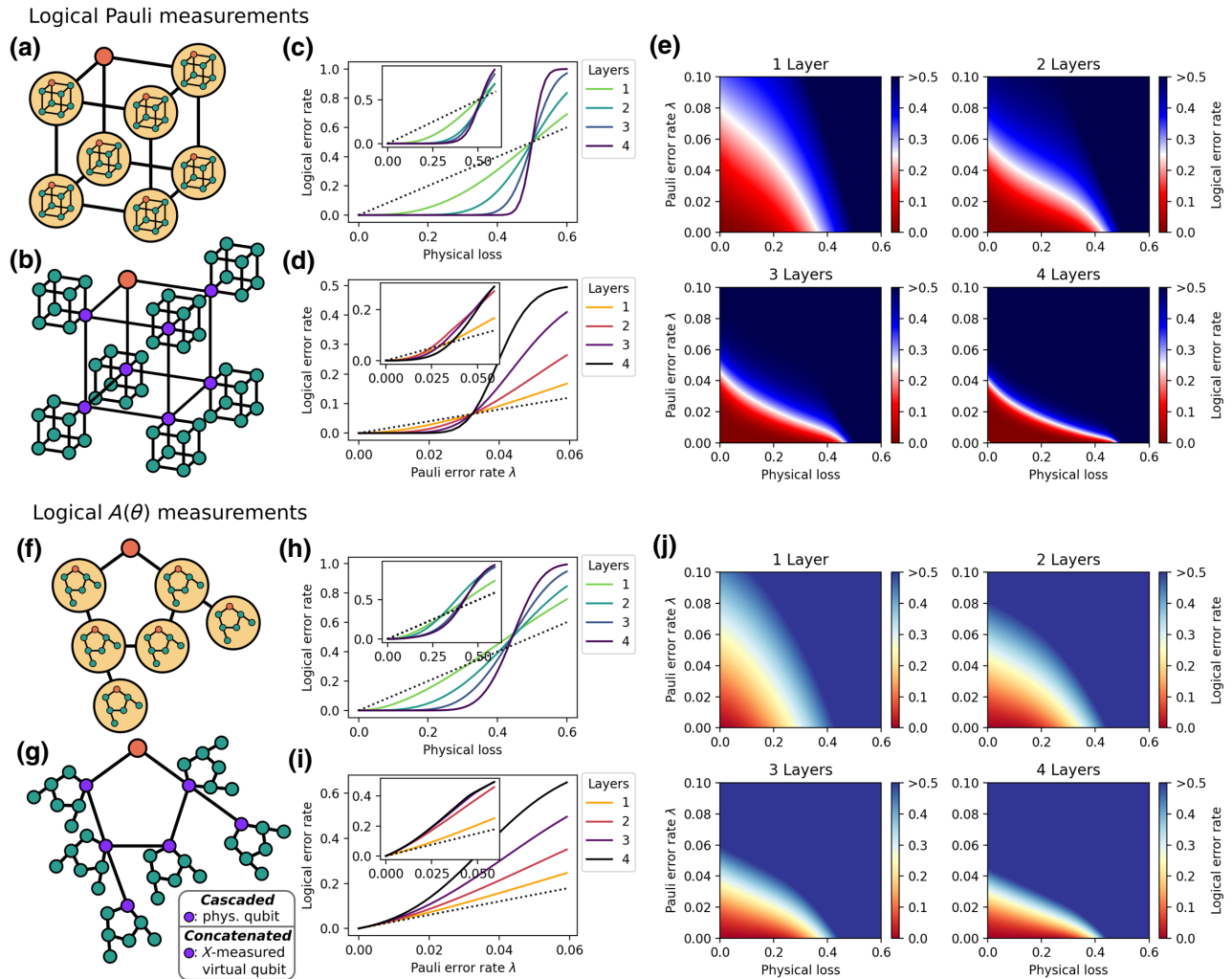
FIG. 7. Performance of cascaded and concatenated codes. (a) Modular construction using the cube graph code, and (b) the resulting progenitor graph state with two layers. The qubits in layer 1, shown in purple, are physical for the cascaded construction, and virtual X-measured qubits for the concatenated one. (c) Loss-tolerance performance, considering a loss-only noise model, for logical Pauli measurements self-concatenating the cube graph code at different layers, showing a threshold at 50% loss. The performance for equivalent cascaded constructions is shown in the inset. (d) Performance for error tolerance, with an IID error-only model, showing a threshold at $\lambda \simeq 3.2\%$ (e) Overall fault probability in the presence of both noise types for concatenations of up to depth 4, showing the emergence of a threshold curve. (e)–(j) Equivalent plots for the decorated pentagon code for arbitrary basis measurements.

[the set B in Fig. 2(c)]. Such measurements are effectively disentangling deeper layers, analogously as in tree graphs [6]. The asymmetry between Z and non-Z bases can thus be attributed to the geometry of the stabilizers in the full graph; X- and Y-type stabilizers penetrate between layers, whereas Z-type stabilizers do not.

Using the properties described above, we can perform the decoding process for cascaded graphs recursively using only the properties of unit graphs at different layers in the cascade. We report such recursive functions, for decoding both losses and errors, in Appendix B 1. In Figs. 7(c) and 7(d) (insets) we report an example of improved logical losses and error rates obtained by cascaded graph structures. Here the unit graph is taken to be the

smallest graph we identified in Sec. III with tolerance to both loss and errors for logical Pauli measurements—the cube progenitor graph.

B. Concatenated graphs

Modular extensions of graph codes can also be performed via graph-code concatenation, where each code qubit is itself encoded in another code—a standard approach in QEC. Concatenation of graph codes can be described by simple graph operations and be used for constructing concatenated quantum codes of increasing size [32]. A concatenated graph code can be also easily described starting from the cascaded construction of the

previous section: it corresponds to considering every qubit in intermediate layers as virtual qubits measured in the X basis and with $+1$ outcome obtained. The measurement-based decoding procedures can also be performed similarly in a recursive fashion, with the only difference that now all measurements performed on intermediate layers are indirect and direct measurements are only performed on the qubits in the deepest layer (see Appendix B 2 for details). Note in fact that virtual qubits do not have to exist in practice and are only useful in describing the concatenated graph; only the lowest depth qubits are physical. We show examples of concatenated codes in an encoding is shown in Figs. 7(b) and 7(g) for self-concatenations of the cube (Steane) and decorated pentagon graph codes.

In Figs. 7(c)–7(e) we report the performance for concatenated cube graph codes in the presence of losses and errors, for up to 4 layers of concatenation when performing Pauli-basis measurements. We observe a threshold appearing for losses at 50%, saturating the bound set by the measurement complementarity principle (see Appendix C), and an error threshold of $\lambda = 3.2\%$. Figures 7(h)–7(j) show equivalent plots but for arbitrary logical measurements concatenating the decorated pentagon graph. As the decoding procedure described above can easily incorporate the concatenation of different unit graphs at different layers, we can optimise the combination of unit graphs to obtain higher noise tolerance at various number of code qubits. In Fig. 8 we plot the results of such optimization for concatenated graphs at different loss levels, for both logical Pauli and arbitrary measurements. The optimization was performed by directly testing all combinations for up to four layers of unit graphs from the set of all loss-tolerant graphs we identified from the analysis in Sec. II. For comparison, we also report the performance of optimized tree graphs from Ref. [6]. Already in the regime with few tens of qubits, we see orders of magnitude improvement in the logical loss of the optimized concatenated graphs against tree graphs for the tested physical transmission rates in the 0.7–0.95 range.

V. LOGICAL GRAPH-STATE FUSIONS

The techniques described in previous sections for single-qubit logical measurements can also be used to analyze another key operation for photonic measurement-based approaches: *fusion gates* [4,34,35]. Introduced in Ref. [34], they are probabilistic two-qubit entangling gates that can be implemented using simple linear-optical circuits, and have the effect of joining two photonic graph states with a destructive measurement on two photons, one from each graph. Standard fusion operations, implementable with simple linear circuits and no ancillary resources, succeed with a probability of 50% [34], which can be boosted by using ancillary photons [36–38]. In particular, Ref. [36] shows that using $2^m - 2$ entangled ancillary photons it is
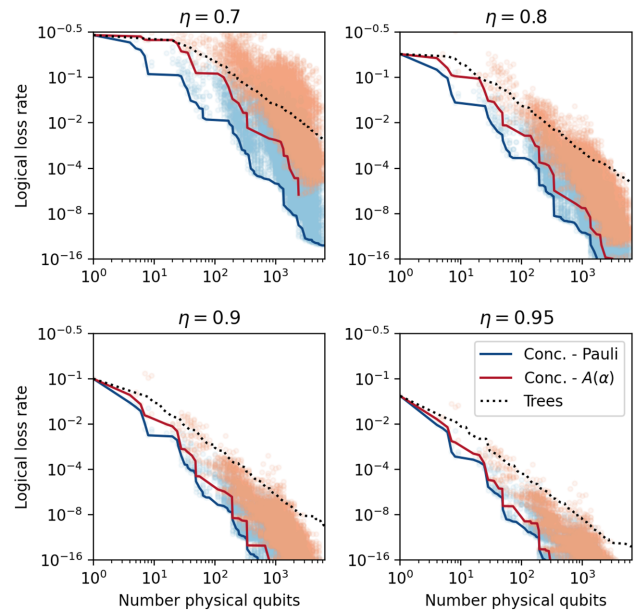

FIG. 8. Resource scaling for optimized concatenated graph codes for logical Pauli (blue) and arbitrary $A(\alpha)$ (red) measurements, compared with the performance of optimized tree graphs (dashed black) from Ref. [33]. For tree graphs, the worst-case logical loss rate of a Pauli measurement (Y basis) is the same as for $A(\alpha)$. Semitransparent points correspond to logical losses for all the combinations of concatenated graphs tested in the optimization procedure, and solid lines correspond to the optimized scaling from the best concatenation configurations obtained. The numerical scaling data, which considers exclusively losses, is reported for different values of the per-photon transmission η. Note the scale for all plots is log versus loglog.

possible to achieve a success probability $(1 - p_{\text{fail}})\eta^{1/p_{\text{fail}}}$, where $p_{\text{fail}} = 2^{-m}$ is the probability of gate failure and η is the transmission. In operator terms, a successful fusion measurement retrieves two parity measurements for the operators XX and ZZ. If the gate fails, only one of the two outcomes is available, and single-photon operations can be used to choose it to be ZZ or XX, while the other outcome is erased. If either of the qubits is lost, the gate fails completely and neither operator is recovered. The two mechanisms of qubit loss and gate failure are thus inequivalent and with different consequences for the growth of clusters. Graph codes can be used to make the fusion of the encoded logical qubits robust against both mechanisms [4,39–42]. A logical fusion of two encoded qubits is a measurement providing joint parity checks $\overline{XX}$ and $\overline{ZZ}$. In contrast to logical single-qubit measurements, this operation requires physical fusion gates between qubits from the two codes, i.e., physical fusion gates. Nevertheless, as we show, these can be readily included with the techniques developed in previous sections. We consider two different strategies for it, as illustrated in Fig. 9(a) and described below.
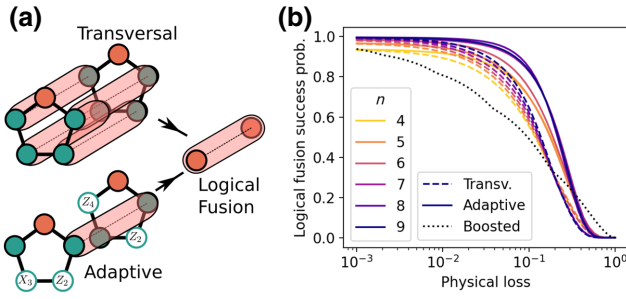
FIG. 9. Logical fusion measurements. (a) Two approaches to perform logical fusion operations are to perform physical fusion transversally in a ballistic manner (top), or using an adaptive strategy requiring only a single successful physical fusion and with single-qubit measurements on the remaining qubits (bottom). (b) The probability of logical fusion $\overline{p}_{\text{succ}}$ is plotted for the adaptive (solid lines) and transversal (dashed) approaches, for varying code sizes. Standard physical fusions with 50% success probability are considered between physical qubits. For comparison, we also report the optimal performance of boosted fusion measurements (black dotted line) using 2^{m-1} ancilla qubits for $m \leq 8$.

A. Transversal physical fusions

Considering two identical graphs encoding the logical qubits to be fused, we first consider a ballistic method where physical fusions are attempted transversally between all code qubits in one graph and the equivalent qubits in the other, as shown in Fig. 9(a). Each physical fusion can be successful, fail, or be erased due to loss of one of the photons with respective probabilities $\eta^{1/p_{\text{fail}}}(1 - p_{\text{fail}})$, $\eta^{1/p_{\text{fail}}}p_{\text{fail}}$, and $1 - \eta^{1/p_{\text{fail}}}$. Once all transversal fusions are performed, the logical fusion is successful if the obtained operators can generate both $\overline{XX}$ and $\overline{ZZ}$, fails if only one of them can be generated, or is completely lost if neither of them can. We numerically calculate the probability for each of these three logical outcomes by considering all possible combinations of the three outcomes from fusing all n pairs of code qubits in the graphs. The total logical fusion success probability is obtained by summing the probability associated to all combinations that lead to a successful fusion, and similarly for the logical failure and logical loss probability. We allow the measurement recovered on physical fusion failure to be chosen independently for each qubit, which is precompiled using maximum likelihood before runtime to maximize the probability of successful fusion. We report in Fig. 9(b) results for logical transversal fusion on the graphs optimized at subthreshold loss rates for arbitrary single-qubit measurements up to $n = 9$ [see Fig. 5(d)]. Despite the approach being nonadaptive, these small codes present success probabilities that significantly outperform typical boosted fusion schemes, which we also report in the black dashed curve, in terms of loss tolerance. For example, using standard physical fusions with 50% success

probability, we obtain logical fusion measurements with success probability $\overline{p}_{\text{succ}} > 0.8$ for physical loss $\ell = 5\%$ and $\overline{p}_{\text{succ}} > 0.95$ at $\ell = 1\%$.

B. Adaptive physical fusions

A second approach we investigate for logical fusions is an adaptive strategy based on the ideas introduced for performing logical arbitrary basis measurements in Sec. II B. Recall that the SPC identifies pairs of logical operators that anticommute on a single qubit [22], which we denote as output qubit. Taking two copies of a graph code, a logical fusion can be achieved by fusing an identified output qubit with the corresponding qubit in the other graph, followed by single-qubit measurements on the remaining qubits. The decoder for adaptive fusion is similar to that presented in Algorithm 2: the idea, in the progenitor graph picture, is to teleport the virtual input qubits of the two codes into some output qubits, which have been prefused together. Explicitly, a physical fusion measurement is attempted between pairs of output qubits, and once a fusion is successful single-qubit Pauli measurements are attempted sequentially on the remaining qubits of each code, effectively implementing a separate decoder as in Algorithm 2 for each graph. This approach leads to improved loss tolerance compared to the transversal case as, when performing single-qubit measurements, the loss of either qubit in a pair does not erase the information obtainable from the other. The use of single-qubit measurements to increase the loss tolerance of logical fusions has been suggested for tree graphs [42], here we extend to general graphs.

In Fig. 9 we report the performance of the adaptive strategy using the graphs identified in Sec. II B, which are optimized for arbitrary single-qubit measurements up to $n = 9$ [see Fig. 5(d)]. It can be observed that the adaptive strategy generally provides better performance compared to the transversal one and boosted fusions. For example, using standard physical fusions with 50% success probability, we can reach logical fusion success probabilities of $\overline{p}_{\text{succ}} = 0.86$ already at a physical loss of $\ell = 10\%$, and $\overline{p}_{\text{succ}} > 0.99$ at $\ell = 1\%$.

VI. APPLICATIONS

To benchmark the tools we develop for analyzing and optimizing general loss-tolerant graph codes, we investigate how they can be used in two exemplary applications.

A. Optimizing repeater graphs

Repeater graph states (RGSs) have been introduced in Ref. [11] as an approach to making all-optical two-way quantum repeaters in a quantum network. The graph structure originally proposed is shown in Fig. 10(a), and the
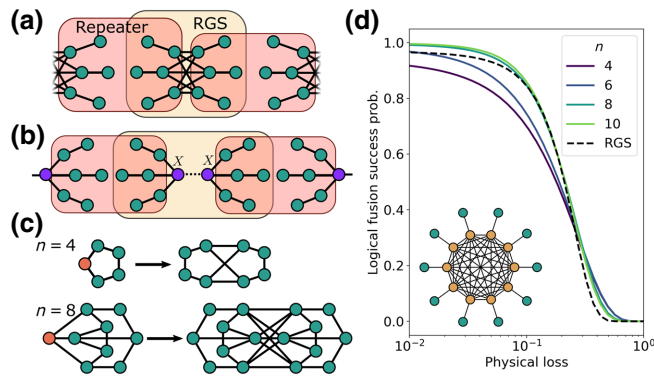
FIG. 10.   Generalized repeater graph states. (a) Operation of the $N = 3$ repeater graph state (RGS, in the yellow box) introduced in Ref. [11]. Single-qubit Pauli and fusion measurements are performed on qubits to fuse each RGS with its neighbors at consecutive repeater stations (red boxes). (b) The progenitor graph construction of RGSs, where logical fusions are performed at each repeater station between progenitor graphs whose input qubits have been joined via a controlled-phase gate. (c) Generalization of RGS states constructed from loss-tolerant graph codes with $n = 4$ and $n = 8$ code qubits. (d) Link success probability as a function of the physical loss rate for graph codes with adaptive nonboosted physical fusions (solid lines), with varying numbers of code qubits, against an instance of the RGS construction with 90 physical qubits ($n = 45$), shown inset. The yellow qubits are logically encoded in a tree graph with branching ratios [4, 1].

repeater protocol works by transversely fusing the left-ward leaf (i.e., single-edged) qubits from a repeater station with the right-ward leaf qubits from the previous station. The inner qubits are measured in X if the associated leaf is the first to be successfully fused, or otherwise in Z to remove unsuccessful or redundant fusions [11].

This protocol can also be interpreted as sequences of logical fusion operations described in Sec. V, and can be readily analyzed in the progenitor graph picture as depicted in Fig. 10(b). The left-ward and right-ward qubits each correspond to physical qubits in a graph code encoding a single logical qubit. Entanglement swapping between successive repeater stations simply corresponds to a logical fusion operation using the right-ward and left-ward codes. Within the repeater graph, the transmission of the logical information between the left-ward and right-ward codes can be simply described, again using the progenitor graph picture, as adding a link (i.e., a controlled-phase gate) between the input qubits of the left-ward and right-ward codes and then measuring both of them in X to transmit the encoded logical qubit between left and right [6]. In practice, the inputs are just treated as virtual qubits, and we directly consider the total repeater graph, i.e., the graph obtained after the controlled-phase and the X measurements are performed. The probability to successfully transmit between two consecutive stations thus simply

corresponds to the logical fusion success probability of the underlying code as analyzed in Sec. V.

For the repeater graph considered in Ref. [11], it is easy to see that, up to local operations [43], it corresponds to using tree-graph codes with a branching ratio $[N/2, 1]$ as both left-ward and right-ward codes [see Fig. 10(b)], with N the branching of the repeater graph [11]. However, as discussed in the previous sections, tree codes are suboptimal for loss tolerance, and better performance can be obtained using codes optimized for logical fusion success probability. Using the construction presented above, we show in Fig. 10(c) the repeater graphs obtained from two optimized codes for logical fusion with $n = 4$ (the pentagon graph) and $n = 8$. The metric used in this optimization is the total success probability of the logical fusion, i.e., the probability to obtain both $\overline{XX}$ and $\overline{ZZ}$. Their performance with adaptive fusion strategies is reported in Fig. 10(d), in which we also show for comparison the performance of standard RGSs. We see that comparable link generation probability can be achieved using general graph states with vastly fewer physical qubits, compared to traditional tree-based encodings, showing that our tools can bring significant improvements in the design of all-optical repeater schemes.

B. Fusion-based fault-tolerant schemes

Fusion-based quantum computation (FBQC) is a variant of measurement-based quantum computing where the computation is performed via probabilistic fusion gates between separate resource states rather than single-qubit measurements on large entangled cluster states [4]. It has been recently introduced as a convenient picture to describe photonic quantum computation as it facilitates a direct description of probabilistic fault-tolerant architectures fusing small resource states, enabling a simple treatment of failed fusion operations and qubit loss. However, in the constructions from the original proposals [4], the per-photon loss thresholds are limited to <1%, also requiring highly boosted fusions. To improve them, concatenating qubits with a (2,2) Shor code was proposed, whereby a loss-tolerance threshold of 2.7% per photon is achieved for boosted physical fusions with 75% success probability. To obtain better performances, we can use the techniques developed in previous sections to consider concatenating resource states with more general graph codes, as shown in Figs. 11(a) and 11(b).

In FBQC, the XX and ZZ parity measurement outcomes obtained from qubit fusions are used to construct the primal and dual syndrome graphs of a RHG lattice. The probability of logical error of the topological qubit depends on the probability that XX and ZZ measurement outcomes are erased. A difference with the logical fusion analyses performed in previous sections is that now we need to differentiate an unsuccessful fusion due to gate failure, where
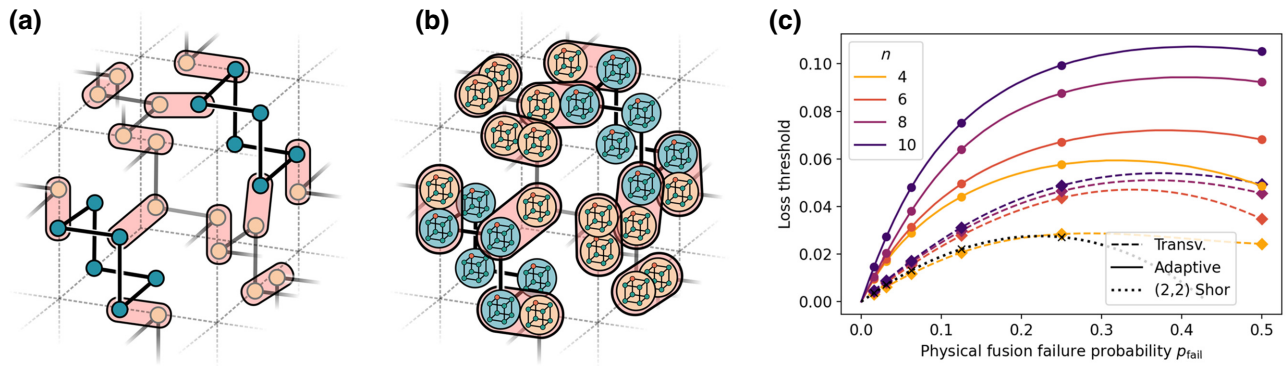
FIG. 11. Fusion-based fault tolerance. (a) Construction of a fault-tolerant fusion network for FBQC via fusing six-qubit hexagonal resource state, from Ref. [4]. (b) Concatenation of the fusion network with graph codes to enable logical fusions tolerant to loss and failure. (c) Per-photon loss-tolerance thresholds for concatenations with optimized graph codes of varying sizes, using the adaptive decoding strategy (solid lines) and the transversal strategy (dashed). The performance of the (2,2) Shor encoding presented in Ref. [4] is also shown as the black dotted line.

only one of XX and ZZ is erased, with the unsuccessful case due to the loss, where both outcomes are erased. In fact, randomizing the erased outcome in failed cases as in Ref. [4], which for general graph codes can be done via local Clifford operations, a failed fusion still has 50% chance to provide the outcome required for either the primal or dual syndrome graph. Therefore, in unsuccessful cases, we seek to enhance logical failure instead of logical loss, leading to a different optimization strategy.

To focus on a specific architecture, we consider the fault-tolerant fusion network constructed from fusing six-qubit hexagonal resource states from Ref. [4] [see Fig. 11(a)], which has the highest measurement erasure threshold amongst the reported FBQC schemes, i.e., 12%. The loss threshold per photon is then the threshold at which $\overline{XX}$ and $\overline{ZZ}$ erasure probabilities are simultaneously suppressed below 12%. Such thresholds in general also depend on the failure rate of the physical fusions employed, as boosting the success probability of the gate requires an increasing number of ancillary photons, none of which are permitted to be lost. It is therefore important to consider the trade-off between fusion failure rates and loss tolerance. These trade-offs result in concave curves for loss thresholds as a function of physical failure rates [4], as the ones shown in Fig. 11(c). In our analysis we optimize graphs considering only standard physical fusion gates, i.e., with 50% success probability, but for completeness we report the performance also for boosted cases. By parallelizing the optimization procedure using the Blue-Crystal high-performance computing cluster, we optimize for graph states with up to $n = 10$ (i.e., 11-qubit progenitor graphs), considering both transversal and adaptive fusions. The identified graphs, chosen to maximize the resultant loss-tolerance threshold, are shown in Appendix F, provide the loss thresholds reported in Fig. 11(c). For the adaptive approach, the thresholds reach 10.5% for qubit graph codes

with $n = 10$, considering nonboosted physical fusions, and 4.9% for the transversal approach.

VII. DISCUSSION

We have shown how developing methods to analyze the measurement-based loss tolerance, as well as error-correction properties, for arbitrary graph states can provide logical qubits with significantly higher noise tolerance and fewer physical qubits. This is observed both for modules with $\simeq 10$ qubits and in the asymptotic regime where orders of magnitude improvements are observed with respect to tree graphs. An immediate implication of these results is to show that most of the graph modules currently considered in various photonic-based applications, such as tree-based encodings for one-way and two-way quantum repeater protocols [11–13,44] and Bacon-Shor codes for logical fusions in FBQC architectures [4], are suboptimal. Significant improvements can be obtained by using graphs optimized for the targeted functionality. We illustrate these advantages for a few applications in Sec. VI, but expect it to be relevant to improve a large part of photonic quantum applications based on graph states. To this scope, we make the Python code utilized for all the analysis in this work freely accessible [45]. As an example, the per-photon-loss threshold of 10.5% for fault-tolerant FBQC, obtained considering only standard nonboosted fusion gates, is a significant improvement with respect to the previous 2.7% value with boosted fusions from Ref. [4], potentially bringing fault tolerance much closer to the capabilities of near-term photonic hardware. Moreover, this threshold is obtained considering the fusion network construction from Ref. [4] based on fusing six-qubit hexagons as resource states, and we expect it to improve further by developing fusion networks with higher tolerance to fusion erasure [46].

Technologies that are in principle well suited for the generation of graph codes include all-optical approaches, which when equipped with feed-forward and multiplexing could generate graph resource states deterministically [34,35,47], and approaches based on quantum emitters, where photonic entanglement can be directly generated via spin-photon interfaces [48–50]. In particular, high-fidelity spin-photon systems have been recently developed in a variety of platforms, including quantum dots, superconducting circuits, atoms in optical cavities, and N-V centers [51–57], with demonstrations of deterministic generation of graph states with up to 14 photons [57].

The tools developed here allowed us to identify loss-tolerant graph codes with minimal requirement in terms of the number of qubits, and can be readily adapted to incorporate hardware-specific restrictions and error models. We expect such capabilities to be significantly valuable in developing near-term experiments targeting loss tolerance in the photonic platform. Such demonstrations will provide truly loss-tolerant photonic qubits, a milestone yet to be achieved that promise to unlock important opportunities for scaling photonic quantum technologies.

APPENDIX A: DECODERS

1. Loss-only decoding

The decoding strategy for arbitrary basis measurements on graph codes implemented in this work is outlined in Algorithm 2, with small modifications to that algorithm for Pauli basis or fusion measurements. The general strategy is as follows. Firstly, all possible measurement are initialized. For the case of Pauli measurements, this is simply the set of logical operators $\overline{\mathcal{L}}$, whereas for $\overline{A}(\theta)$ or fusion measurements, it is an operator satisfying the SPC [22] on the progenitor graph for any choice of output qubit, combined with physical $A(\theta)$ or fusion measurements on that qubit. Inspired by Ref. [22] we construct measurements from only *nontrivial* stabilizers and logical operators, i.e., those which cannot be decomposed into a smaller weight operator multiplied by a stabilizer with nonoverlapping support, reducing the number of operators to consider. Then the optimal first measurement is determined according to

a cost function—typically by choosing the measurement with the lowest weight, and selecting a random qubit from its support. On attempting the measurement, the outcome is recorded (± 1 or *null* for a lost qubit), as well as the attempted basis. Now the set of available measurements is updated in response to the outcome, according to Eq. (2). These three steps are repeated until a measurement has succeeded or no possible strategies remain. Performing this search for every configuration of lost qubits constitutes building a decision tree for the decoder offline so that real-time decoding is simply a look-up table, which is queried up to n times during measurement of an n-qubit graph code. The tree is built using a depth-first search, and the termination conditions of the decoder mean that not all loss configurations need be examined. Every leaf of the tree is an outcome of the decoding procedure, with an associated set of measured qubits A, lost qubits B, and an outcome $\in$ {success, fail}. If we call the set of successful leaves as Q, the probability of a successful measurement for the graph state G is

$$\overline{\eta} = \mathcal{F}(\eta) = \sum_{q \in Q} P_q(\eta) = \sum_{q \in Q} \eta^{|A_q|}(1 - \eta)^{|B_q|}. \quad \text{(A1)}$$

This gives us analytic expressions for the effective transmission rates for various basis measurements on qubits encoded in graph codes.

2. Loss and unitary errors

At a qubit level, mitigation of unitary errors is done by measurement of code qubits remaining after the target measurement has been completed—these are additional resources that we leverage to gain more information about the target measurement outcome. The decoder is accordingly adapted such that we no longer terminate after the loss-only decoder is finished, but instead determine which additional stabilizer measurements can be implemented to check the obtained outcome. When the optimal check measurements have been identified, they are attempted in succession, and as before after each the strategy is updated according to its success or failure. This effectively extends the decision tree, reflecting the increased computational overhead of simultaneous error and loss correction.

Choosing optimal check measurements is a nontrivial task. From the set of remaining valid stabilizers, a set of valid check operators must commute qubitwise with one another, and with the set of measurements already performed. To choose a check set, we use heuristic methods, and pick the largest qubitwise commuting set with the greatest overlap with the target measurement. It should be noticed that this is not necessarily optimal.

The error-tolerant performance of this approach is determined numerically. For each successful leaf of the decision tree, there is an associated target measurement, and a (possibly empty) set of check measurements. As detailed in

the main text, we consider the phenomenological noise model, in which Pauli operators are randomly applied to each code qubit with probability λ. This may result in flipped measurement outcomes, which is a logical error on the measurement. To find the probability of a logical error, we consider all configurations of measurement error on all code qubits, which are binary strings of length n, denoted $\mathbf{e}$. Their probability is calculated, for the phenomenological model the probability of a flipped Pauli-basis measurement is $\epsilon_{\text{Pauli}} = 2\lambda$, as there are two anticommuting Pauli errors, and for an arbitrary basis measurement $\epsilon_A = 3\lambda$, so $P_{\mathbf{e}} = \sum_{i=1}^n \epsilon_i^{e_i}(1-\epsilon_i)^{(1\oplus e_i)}$. The syndrome is found from the outcomes of the check measurements, $\text{Synd}(\mathbf{e}) = \{\pi(\mathbf{e}_{C_j})\}$, where j indexes the jth check operator, and $\pi(x)$ denotes the parity of the string x. We also determine whether a logical error occurred on the target measurement, from the parity of $\mathbf{e}$ on the target measurement. From this, we can find the most probable error $P_{\max}$ on the target given a particular syndrome, and so given a particular syndrome (which occurs with known probability) we correct for the most likely error pattern, succeeding with probability $P_{\max}$. Summing over all syndromes gives the total error rate. This process needs to be done for each successful leaf of the decision tree, as the target and check measurements will differ, with the overall performance of the graph being the logical error probability for each leaf weighted by the probability of obtaining that configuration [Eq. (A1)]. Note that for logical arbitrary basis measurements we need to know the signs of two measurements to correctly decode the result, adding additional difficulty to the decoding process. Again, the computationally expensive parts of this decoding can be done offline, resulting in look-up table runtime costs.

APPENDIX B: ADAPTING MEASUREMENT PATTERNS IN CASCADED AND CONCATENATED GRAPHS

1. Cascaded graphs

Suppose we want to implement a local Pauli measurement pattern M on a graph G, in order to implement either an indirect Pauli measurement or a SPF-based teleportation strategy. How do the required measurements change when another graph is appended to each qubit of G (except the input)? The strategies outlined in this work are constructed from stabilizers, so we can consider how stabilizers from G_U are modified by moving to the cascaded graph. By decomposing the stabilizer into products of generators as outlined in Sec. I A, we see that if $b_t = 0$, the modified measurement on the cascaded graph can have no weight on G_L, as G_L is only adjacent to qubit t. $b_t = 0$ additionally implies $M^{[t]} = Z$ or $\mathbb{1}$, so Z measurements do not need to be modified when switching to the cascaded graph. This is intuitive from the graphical perspective as well, the Z_t measurement deletes that vertex from the graph,

and G_L is disconnected from G_U. For the same reasons, indirect Z_t measurements, denoted $Z_t^{(\text{ind})}$, can be performed by measuring qubits in G_L only, leading to the improved loss tolerance of cascaded graphs. If instead $b_t = 1$ and $M^{[t]} = X$ or Y, the corresponding stabilizer in the cascaded graph penetrates in to G_L. The measurement pattern on the graph G_L is $\prod_{j\in\mathcal{N}(t)} Z_j$, an indirect X measurement on qubit t. Any X measurement on an intermediate layer code qubit therefore requires an additional indirect X basis measurement on deeper qubits, to disentangle them from the graph. This can be multiplied by any stabilizer of $G_L + t$ that does not include the generator $K_t(G_L + t)$, the generator corresponding to qubit t in the graph ($G_L + t$), to ensure the measurements in G_U are unaffected. If we multiply by an odd number of stabilizer generators in the neighborhood of t, this alters the required measurement on qubit t. For an arbitrary basis measurement, we can derive the measurement update rules by inspecting the changes to the stabilizers involved in constructing the measurement pattern (as outlined in Sec. II B). At least one of these stabilizers will include the generator of the output qubit, so in cascading this generator is modified as for X or Y measurements outlined above. These updates are then given by Eq. (B1).

$$
\begin{aligned}
X &\longrightarrow X_t X_t^{(\text{ind})}(G_L) \text{ or } Y_t Y_t^{(\text{ind})}(G_L), \\
Y &\longrightarrow Y_t X_t^{(\text{ind})}(G_L) \text{ or } X_t Y_t^{(\text{ind})}(G_L), \\
Z &\longrightarrow Z_t \mathbb{1}_{G_L} \text{ or } \mathbb{1}_t Z_t^{(\text{ind})}(G_L), \\
A &\longrightarrow A_t X_t^{(\text{ind})}(G_L) \text{ or } \tilde{A}_t Y_t^{(\text{ind})}(G_L),
\end{aligned}
\tag{B1}
$$

where $\tilde{A}$ represents a modified basis. For the cases of X, Y, and A the two options are incompatible, so one must choose which to attempt in advance. It is clear to see that both the direct and indirect measurements must succeed for one of these measurements to be successfully performed. The probability of successfully measuring a qubit in basis $M \neq Z$ becomes

$$
\eta_{M,\text{casc}}^{(k)} = \eta \mathcal{F}(\mathbf{r}^{(k+1)}; G_k, M), \tag{B2}
$$

where $\mathcal{F}(\mathbf{r}^{(k+1)}; G_k, M)$ gives the probability of successfully performing a logical M basis measurement as a function $\mathbf{r}^{(k+1)} = \{\eta_X^{(k+1)}, \eta_Y^{(k+1)}, \eta_Z^{(k+1)}, \eta_A^{(k+1)}\}$ on graph G_k at depth k in the cascade. η is the physical transmission probability. This function can be calculated analytically by considering the small unit graphs that make up the cascade. For a Z-basis measurement, the two options in Eq. (B1) are compatible, so we can try both. The logical transmission is then calculated via

$$
\eta_{Z,\text{casc}}^{(k)} = \eta + (1-\eta)\mathcal{F}(\mathbf{r}^{(k+1)}; G_k, Z). \tag{B3}
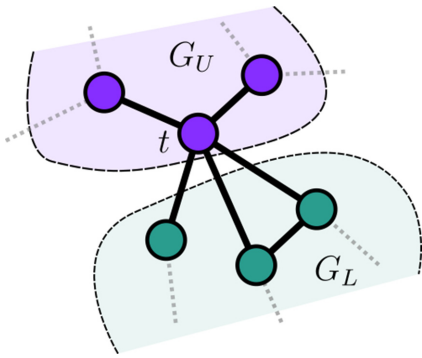$$

FIG. 12. A Pauli measurement on an intermediate-layer qubit t has additional requirements when we move to a cascaded graph. t is a "bottleneck" qubit between the upper and lower graphs G_U, G_L. We can find the properties of the casaded graph by considering the performance of the graphs G_U, $G_L + t$ for SPF and indirect Pauli measurements.

2. Concatenated graphs

In concatenated graphs the picture is similar, except now every every "bottleneck" qubit is a virtual qubit—it has already been measured in the X basis, and the $+1$ outcomes obtained. The only measurement patterns than can be kept are then those compatible with X_t measurements (using the same naming conventions as in Fig. 12). For Pauli-basis measurements, these are simply read off from Eq. (B1). For the arbitrary basis measurement the situation is slightly different, as now we choose an output qubit in the deepest layer of the concatenation. As described in the main text, we can think of this as teleporting to a depth k qubit, and then instead of measuring it in the arbitrary basis, teleporting it deeper into the concatenation via the already performed X measurement of the virtual bottleneck qubits. Hence one can think of measuring the virtual qubit in an arbitrary basis by performing a teleportation measurement pattern on the qubits in the $k + 1$th layer.

$$
\begin{aligned}
X &\longrightarrow X_t X_t^{(\text{ind})}(G_L), \\
Y &\longrightarrow X_t Y_t^{(\text{ind})}(G_L), \\
Z &\longrightarrow Z_t^{(\text{ind})}(G_L), \\
A &\longrightarrow A_t^{(\text{ind})}(G_L).
\end{aligned}
\tag{B4}
$$

It can be seen directly from these expressions that the effective transmission parameters for virtual qubits at depth k is simply the probability of performing indirect measurements on qubits in the layer below, so can be recursively calculated according to the expression

$$
\eta_{M,\text{conc}}^{(k)} = \mathcal{F}(\mathbf{r}^{(k+1)}; G_k, M).
\tag{B5}
$$

This form enables us to recover true loss-tolerance thresholds for MBQEC under code concatenation if the $\mathcal{F}$ are
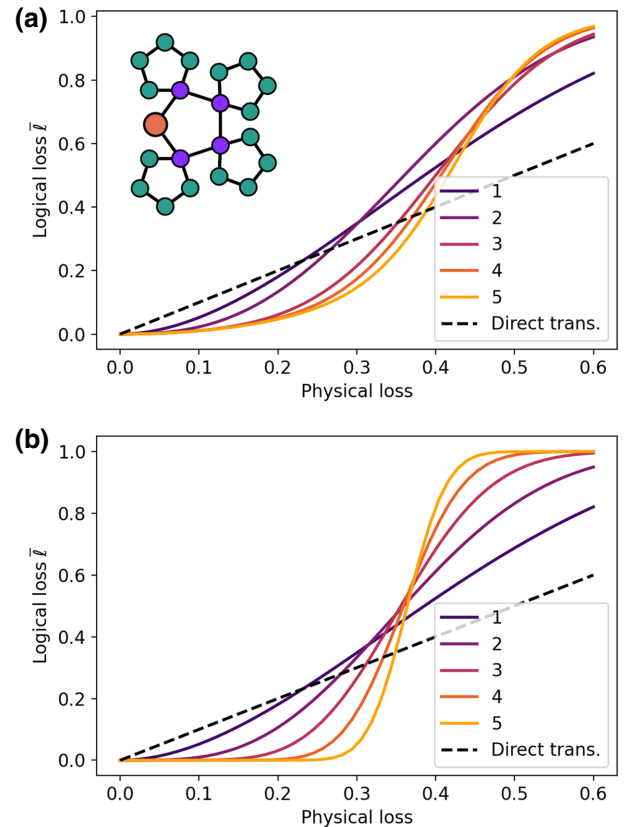

FIG. 13. Cascaded and concatenated graphs. (a) Logical against physical loss rates for arbitrary measurements on cascaded pentagon graphs. Inset is a depth-2 cascade with 20 physical qubits (purple qubits are physical). (b) shows the analogous performance for concatenated pentagons, for which the depth-2 concatenation has 16 physical qubits (purple qubits virtual).

the same for all measurements in the target pattern. This is seen in the Steane code thresholds for MBQEC Pauli-basis measurements discussed in the main text, where $\mathcal{F}(\mathbf{r}; \text{Steane}, X) = \mathcal{F}(\mathbf{r}; \text{Steane}, Y) = \mathcal{F}(\mathbf{r}; \text{Steane}, Z)$.

In Fig. 13 we compare cascaded and concatenated performance for an exemplary graph, here chosen to be the smallest progenitor graph with loss tolerance for arbitrary basis measurements—the pentagon graph. Both methods suppress logical error rates, but we see that the concatenated graphs have lower loss rates and fewer code qubits. In this work we have not considered how to physically realize these modular code constructions—it may be that the concatenated approach suffers from greater overhead during the preparation stage. This would be an interesting and useful avenue for further investigations.

APPENDIX C: MEASUREMENT COMPLEMENTARITY PRINCIPLE

By the uncertainty principle, one cannot perform measurements in different bases on a single qubit
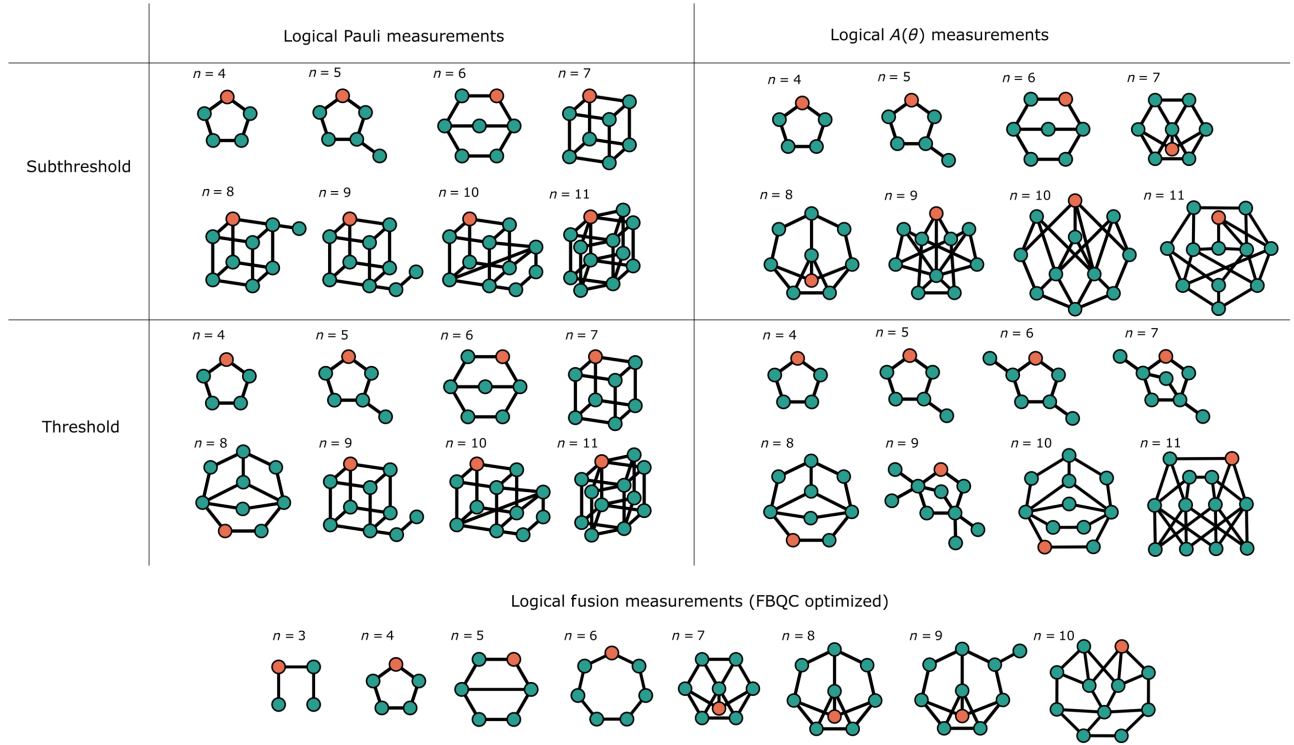
FIG. 14. The optimised graphs for loss-tolerant logical qubit measurements (top), for break-even and subthreshold regimes, and optimized graphs for logical fusion measurements (bottom).

simultaneously, which has implications for the loss-tolerance thresholds of quantum codes, in particular that the thresholds may not exceed 50%. Consider an $[[n, 1, d]]$ stabilizer code, with stabilizer group $\mathcal{S}$ and logical operators $\overline{X}$, $\overline{Z}$. If there exists a bipartition of the physical qubits into sets A_1, A_2 such that there exist logical operators whose support is restricted to each set $\Sigma(\overline{X}) \subseteq A_1$ and $\Sigma(\overline{Z}) \subseteq A_2$, the two logical operators would be simultaneously measurable, violating the uncertainty principle. This shows that every pair of logical operators shares support on at least one qubit, $\Sigma(\overline{X}) \cap \Sigma(\overline{Z}) \neq \emptyset$. One could imagine losing the set A_2 of physical qubits to a third party. If either party is able to perform a measurement using only their qubits, the above argument necessitates that the other party can only recover the same measurement on their set. To relate this to loss thresholds, we need to distinguish between the *break-even point* and the *threshold*. The break-even point refers to the point at which $\overline{\ell}(\ell_b) = \ell_b$ for a given code, whereas the threshold l^* denotes the loss below which code concatenation increases the probability of measurement success. Given a deep concatenation, the success probability in the subthreshold regime approaches 1. The above argument places no restriction on the break-even point of the codes, but requires that for any two different measurements P and Q, the loss-tolerance thresholds must satisfy $\ell^*[P] + \ell^*[Q] \leq 1$—referred to as the measurement complementarity principle. Hence, the

threshold for arbitrary basis measurements cannot exceed 50%. This is an analogous restriction to the *gate complementarity principle* [31], which applies to the probability of successfully performing logical gates.

APPENDIX D: LOCAL CLIFFORD EQUIVALENCE

In this work we consider codes up to local Clifford operations, such that if two codes can be transformed in to each other by local Clifford operations they are deemed equivalent. When searching for optimal loss-tolerant codes we utilize the fact that the loss tolerance of a graph code using the decoders implemented here is invariant under these operations. This is seen by examining how graph codes are modified under local complementation, a graph transformation, which inverts the neighborhood of a particular node. Local complementation on a node α of a graph is equivalent to application of the following local unitary [18,23] to the graph state.

$$U_\alpha^{\mathrm{LC}} = \sqrt{-iX_\alpha} \bigotimes_{\beta \in N_\alpha} \sqrt{iZ_\beta}, \quad \text{(D1)}$$

where N_α denotes the neighborhood of qubit α. Consider two graph states related via local complementation $|G^{(2)}\rangle = U_\alpha^{\mathrm{LC}}|G^{(1)}\rangle$. These are the progenitor graphs of two locally equivalent graph codes. We show here that the loss

tolerance of these graph codes is identical. Upon conjugation with U_α^{LC} the Pauli operators of a qubit q in the graph transform as

	X	Y	Z	$\mathbb{1}$
$q = \alpha$	X	$-Z$	Y	$\mathbb{1}$
$q \in N_\alpha$	$-Y$	X	Z	$\mathbb{1}$
$q \notin N_\alpha$	X	Y	Z	$\mathbb{1}$

This transformation can be used to readily verify the stabilizer generators of $G^{(1)}$ transform in to the generators of $G^{(2)}$. The loss tolerance of $G^{(1)}$ under logical measurements in an arbitrary basis is determined by the set of valid measurement patterns $\mathcal{M} = \{M\}$, where $M = S_1^{(1)} \cup S_2^{(1)}$, such that the two stabilizers anticommute on the input and output, and commute on all other qubits. Each stabilizer transforms under local complementation according to the above table, such that the qubit support is invariant, and any pair of operators, which are (anti)commuting on a particular qubit remain (anti)commuting on that qubit. This means $S_1^{(2)} \cup S_2^{(2)}$ is a valid measurement pattern on $G^{(2)}$, where $S_j^{(2)} = U_\alpha^{\text{LC}} S_j^{(1)} (U_\alpha^{\text{LC}})^\dagger$, and there is a bijective relation between measurements that perform measurement-based teleportation on graph states. Furthermore, two measurement patterns that were initially compatible remain so, and the success probability is thus conserved.

In general, a local Clifford operation preserves the bitwise commutativity of two Pauli operators, and the compatibility of two measurement patterns is determined by their bitwise commutation properties, so this argument applies similarly to measurements in Pauli bases, and to logical Fusion measurements. For Pauli basis measurements, a local complementation of the progenitor graph may transform a logical Pauli operator in to a different logical Pauli, indeed the locally equivalent graphs depicted in Fig. 3 perform differently to one another in each basis. However, the success probability will be conserved when averaging over Pauli bases, as all logical Paulis that are the same type in one graph will also be the same type in the locally equivalent sibling. The caveat to this invariance is in the decoder implementation. The decoder may make arbitrary choices between equally "good" measurement strategies, biasing the probability of performing measurements in particular bases. A decoder based on heuristic methods may therefore not choose corresponding strategies in locally equivalent graphs, so while their optimal loss tolerance is the same, their performance could vary in practice.

APPENDIX E: STABILIZED SPACES UNDER MEASUREMENT AND LOSS

Consider a stabilizer code with stabilizer $\mathcal{S}$ and logical operators $\overline{X}$ and $\overline{Z}$. Suppose we perform a measurement

$M \in \mathcal{P}^n$ on the qubits in the code. The reduced stabilizer group is given by $\mathcal{S}_M = \{S \in \mathcal{S} \mid [S^{[i]}, M^{[i]}] = 0\}$. For any two stabilizers S_1, S_2 in the reduced group, their product $S_3 = S_1 S_2$ is also in the group. From the requirement that $[S^{[i]}, M^{[i]}] = 0$, we obtain that $S^{[i]} = M^{[i]}$ or $\mathbb{1}$. The product $S_1^{[i]} S_2^{[i]} = M^{[i]}$ or $\mathbb{1}$ from the properties of Pauli operators, and we obtain the commutation relations of S_3, $[S_3^{[i]}, M^{[i]}] = [S_1^{[i]} S_2^{[i]}, M^{[i]}] = 0$ The product element S_3 is therefore in the group, and it is closed. The identity element trivially remains in the group, and $-\mathbb{1}$ cannot be added to the group by discarding elements, so the reduced set $\mathcal{S}_M$ forms a stabilizer group. The effect of loss is similar, except for now we retain only stabilizers that act trivially on the lost set, i.e., if $M^{[i]} = \nexists$, $[S^{[i]}, M^{[i]}] \iff S^{[i]} = \mathbb{1}$. From this it is straightforward to see, by using the same argument as above, that the restricted set of stabilizers satisfying this condition form a stabilizer group.

APPENDIX F: GRAPH LIBRARY

The best-performing graphs for near break-even and subthreshold measurements are shown in Fig. 14, for measurements in both Pauli and arbitrary bases. Also shown in Fig. 14 are the graphs that were found to maximise the loss threshold in the FBQC scheme of Ref. [4]. All graphs shown are minimum edge representatives of a class of locally equivalent graphs.

[1] P. Shor, in *Proceedings of 37th Conference on Foundations of Computer Science* (1996), p. 56.

[2] D. Gottesman, Stabilizer codes and quantum error correction, Preprint ArXiv:quant-ph/9705052 (1997).

[3] E. Knill, R. Laflamme, and W. H. Zurek, Resilient quantum computation, Science **279**, 342 (1998).

[4] S. Bartolucci, P. Birchall, H. Bombín, H. Cable, C. Dawson, M. Gimeno-Segovia, E. Johnston, K. Kieling, N. Nickerson, M. Pant, F. Pastawski, T. Rudolph, and C. Sparrow, Fusion-based quantum computation, Nat. Commun. **14**, 912 (2023).

[5] R. Raussendorf, J. Harrington, and K. Goyal, Topological fault-tolerance in cluster state quantum computation, New J. Phys. **9**, 199 (2007).

[6] M. Varnava, D. E. Browne, and T. Rudolph, Loss Tolerance in One-Way Quantum Computation via Counterfactual Error Correction, Phys. Rev. Lett. **97**, 120501 (2006).

[7] R. Raussendorf, D. E. Browne, and H. J. Briegel, Measurement-based quantum computation on cluster states, Phys. Rev. A **68**, 022312 (2003).

[8] M. Hein, W. Dür, J. Eisert, R. Raussendorf, M. Nest, and H.-J. Briegel, Entanglement in graph states and its applications, Preprint ArXiv:quant-ph/0602096 (2006).

[9] T. Rudolph, Why I am optimistic about the silicon-photonic route to quantum computing, APL Photonics **2**, 030901 (2017).

[10] P. W. Shor, Scheme for reducing decoherence in quantum computer memory, Phys. Rev. A **52**, R2493 (1995).

[11] K. Azuma, K. Tamaki, and H.-K. Lo, All-photonic quantum repeaters, Nat. Commun. **6**, 6787 (2015).

[12] J. Borregaard, H. Pichler, T. Schröder, M. D. Lukin, P. Lodahl, and A. S. Sørensen, One-Way Quantum Repeater Based on Near-Deterministic Photon-Emitter Interfaces, Phys. Rev. X **10**, 021071 (2020).

[13] Y. Zhan, P. Hilaire, E. Barnes, S. E. Economou, and S. Sun, Performance analysis of quantum repeaters enabled by deterministically generated photonic graph states, Preprint ArXiv:2209.11430 (2022).

[14] D. Niu, Y. Zhang, A. Shabani, and H. Shapourian, All-photonic one-way quantum repeaters, Preprint ArXiv:2210.10071 (2022).

[15] Z. Li, I. Kim, and P. Hayden, Concatenation schemes for topological fault-tolerant quantum error correction, Preprint ArXiv:2209.09390 (2022).

[16] D. Schlingemann and R. F. Werner, Quantum error-correcting codes associated with graphs, Phys. Rev. A **65**, 012308 (2001).

[17] D. Schlingemann, Stabilizer codes can be realized as graph codes, Preprint ArXiv:quant-ph/0111080 (2001).

[18] M. Hein, J. Eisert, and H. J. Briegel, Multiparty entanglement in graph states, Phys. Rev. A **69**, 062311 (2004).

[19] M. Rossi, M. Huber, D. Bruß, and C. Macchiavello, Quantum hypergraph states, New J. Phys. **15**, 113022 (2013).

[20] A. Paetznick and B. W. Reichardt, Universal Fault-Tolerant Quantum Computation with Only Transversal Gates and Error Correction, Phys. Rev. Lett. **111**, 090505 (2013).

[21] B. J. Brown and S. Roberts, Universal fault-tolerant measurement-based quantum computation, Phys. Rev. Res. **2**, 033305 (2020).

[22] S. Morley-Short, M. Gimeno-Segovia, T. Rudolph, and H. Cable, Loss-tolerant teleportation on large stabilizer states, Quantum Sci. Technol. **4**, 025014 (2019).

[23] M. Van den Nest, J. Dehaene, and B. De Moor, Graphical description of the action of local Clifford transformations on graph states, Phys. Rev. A **69**, 022316 (2004).

[24] L. E. Danielsen and M. G. Parker, On the classification of all self-dual additive codes over GF(4) of length up to 12, J. Comb. Theory Ser. A **113**, 1351 (2006).

[25] J. C. Adcock, S. Morley-Short, A. Dahlberg, and J. W. Silverstone, Mapping graph state orbits under local complementation, Quantum **4**, 305 (2020).

[26] R. Raussendorf and H. J. Briegel, A One-Way Quantum Computer, Phys. Rev. Lett. **86**, 5188 (2001).

[27] M. A. Nielsen, Quantum computation by measurement and quantum memory, Phys. Lett. A **308**, 96 (2003).

[28] B. Eastin and E. Knill, Restrictions on Transversal Encoded Quantum Gate Sets, Phys. Rev. Lett. **102**, 110502 (2009).

[29] S. Bravyi and R. König, Classification of Topologically Protected Gates for Local Stabilizer Codes, Phys. Rev. Lett. **110**, 170503 (2013).

[30] A. M. Steane, Error Correcting Codes in Quantum Theory, Phys. Rev. Lett. **77**, 793 (1996).

[31] N. Nickerson and H. Bombín, Measurement based fault tolerance beyond foliation, Preprint ArXiv:1810.09621 (2018).

[32] S. Beigi, I. Chuang, M. Grassl, P. Shor, and B. Zeing, Graph concatenation for quantum codes, J. Math. Phys. **52**, 022201 (2011).

[33] M. Varnava, D. E. Browne, and T. Rudolph, Loss tolerant linear optical quantum memory by measurement-based quantum computing, New J. Phys. **9**, 203 (2007).

[34] D. E. Browne and T. Rudolph, Resource-Efficient Linear Optical Quantum Computation, Phys. Rev. Lett. **95**, 010501 (2005).

[35] M. Gimeno-Segovia, P. Shadbolt, D. E. Browne, and T. Rudolph, From Three-Photon Greenberger-Horne-Zeilinger States to Ballistic Universal Quantum Computation, Phys. Rev. Lett. **115**, 020502 (2015).

[36] W. P. Grice, Arbitrarily complete Bell-state measurement using only linear optical elements, Phys. Rev. A **84**, 042331 (2011).

[37] F. Ewert and P. van Loock, 3/4-Efficient Bell Measurement with Passive Linear Optics and Unentangled Ancillae, Phys. Rev. Lett. **113**, 140403 (2014).

[38] A. Olivo and F. Grosshans, Ancilla-assisted linear optical Bell measurements and their optimality, Phys. Rev. A **98**, 042323 (2018).

[39] P. Hilaire, L. Vidro, H. S. Eisenberg, and S. E. Economou, Near-deterministic hybrid generation of arbitrary photonic graph states using a single quantum emitter and linear optics, Preprint ArXiv:2205.09750 (2022).

[40] F. Schmidt and P. van Loock, Efficiencies of logical Bell measurements on Calderbank-Shor-Steane codes with static linear optics, Phys. Rev. A **99**, 062308 (2019).

[41] S.-W. Lee, T. C. Ralph, and H. Jeong, Fundamental building block for all-optical scalable quantum networks, Phys. Rev. A **100**, 052303 (2019).

[42] P. Hilaire, E. Barnes, S. E. Economou, and F. Grosshans, Error-correcting entanglement swapping using a practical logical photon encoding, Phys. Rev. A **104**, 052623 (2021).

[43] Formally, the common RGS is an N-qubit fully connected interior, with leaf qubits on each interior qubit [11]. An identically performing graph (with the exact same decoding procedure) is the graph with a "crazy graph" interior, each with a leaf qubit, which is shown in Fig. 10(a). The progenitor graphs of each of these are the graph with $N/2 + 1$-qubit fully connected interior (including the input node) and leaves on each code qubit, and the tree graph with branching ratios [$N/2, 1$], respectively. These progenitor graphs are locally equivalent and thus have identical loss-tolerance performance.

[44] P. Hilaire, E. Barnes, and S. E. Economou, Resource requirements for efficient quantum communication using all-photonic graph states generated from a few matter qubits, Quantum **5**, 397 (2021).

[45] https://github.com/tomjbell/LTdecode.

[46] S. Paesani and B. J. Brown, High-threshold quantum computing by fusing one-dimensional cluster states, Preprint ArXiv:2212.06775 (2022).

[47] S. Bartolucci, P. M. Birchall, M. Gimeno-Segovia, E. Johnston, K. Kieling, M. Pant, T. Rudolph, J. Smith, C. Sparrow, and M. D. Vidrighin, Creation of entangled photonic states using linear optics, Preprint ArXiv:2106.13825 (2021).

[48] N. H. Lindner and T. Rudolph, Proposal for Pulsed On-Demand Sources of Photonic Cluster State Strings, Phys. Rev. Lett. **103**, 113602 (2009).

[49] S. E. Economou, N. Lindner, and T. Rudolph, Optically Generated 2-Dimensional Photonic Cluster State from Coupled Quantum Dots, Phys. Rev. Lett. **105**, 093601 (2010).

[50] M. Gimeno-Segovia, T. Rudolph, and S. E. Economou, Deterministic Generation of Large-Scale Entangled Photonic Cluster State from Interacting Solid State Emitters, Phys. Rev. Lett. **123**, 070501 (2019).

[51] M. H. Appel, A. Tiranov, S. Pabst, M. L. Chan, C. Starup, Y. Wang, L. Midolo, K. Tiurev, S. Scholz, A. D. Wieck, A. Ludwig, A. S. Sørensen, and P. Lodahl, Entangling a Hole Spin with a Time-Bin Photon: A Waveguide Approach for Quantum Dot Sources of Multiphoton Entanglement, Phys. Rev. Lett. **128**, 233602 (2022).

[52] N. Tomm, A. Javadi, N. O. Antoniadis, D. Najer, M. C. Löbl, A. R. Korsch, R. Schott, S. R. Valentin, A. D. Wieck, A. Ludwig, and R. J. Warburton, A bright and fast source of coherent single photons, Nat. Nanotechnol. **16**, 399 (2021).

[53] J. Arjona Martínez, R. A. Parker, K. C. Chen, C. M. Purser, L. Li, C. P. Michaels, A. M. Stramma, R. Debroux, I. B. Harris, M. Hayhurst Appel, E. C. Nichols, M. E. Trusheim, D. A. Gangloff, D. Englund, and M. Atatüre, Photonic Indistinguishability of the Tin-Vacancy Center in Nanostructured Diamond, Phys. Rev. Lett. **129**, 173603 (2022).

[54] I. Schwartz, D. Cogan, E. R. Schmidgall, Y. Don, L. Gantz, O. Kenneth, N. H. Lindner, and D. Gershoni, Deterministic generation of a cluster state of entangled photons, Science **354**, 434 (2016).

[55] N. Coste, D. Fioretto, N. Belabas, S. C. Wein, P. Hilaire, R. Frantzeskakis, M. Gundin, B. Goes, N. Somaschi, M. Morassi, A. Lemaître, I. Sagnes, A. Harouri, S. E. Economou, A. Auffeves, O. Krebs, L. Lanco, and P. Senellart, High-rate entanglement between a semiconductor spin and indistinguishable photons, Preprint ArXiv:2207.09881 (2022).

[56] K. Tiurev, M. H. Appel, P. L. Mirambell, M. B. Lauritzen, A. Tiranov, P. Lodahl, and A. S. Sørensen, High-fidelity multiphoton-entangled cluster state with solid-state quantum emitters in photonic nanostructures, Phys. Rev. A **105**, L030601 (2022).

[57] P. Thomas, L. Ruscio, O. Morin, and G. Rempe, Efficient generation of entangled multiphoton graph states from a single atom, Nature **608**, 677 (2022).

[58] Advanced Computing Research Centre, University of Bristol, http://www.bristol.ac.uk/acrc/.