Quantum Weight Enumerators and Tensor Networks

ChunJun Cao and Brad Lackey©

Abstract—We examine the use of weight enumerators for analyzing tensor network constructions, and specifically the quantum lego framework recently introduced. We extend the notion of quantum weight enumerators to so-called tensor enumerators, and prove that the trace operation on tensor networks is compatible with a trace operation on tensor enumerators. This allows us to compute quantum weight enumerators of larger codes, such as the ones constructed through tensor network methods, more efficiently. We also provide a general framework for quantum MacWilliams identities that includes tensor enumerators as a special case.

Index Terms—Quantum codes, weight enumerators, tensor networks, MacWilliams identity.

I. INTRODUCTION

Quantum error correction is an essential component for scalable quantum computation. In order to construct good quantum codes, many have drawn on topics from various areas of physics such as quantum many-body theory [1], [2], [3], [4], topological quantum field theory [5], [6], topological order [7], [8], and lattice with defects [9], [10], [11], [12].

Tensor networks [13], [14] can efficiently capture the features of complex quantum states and so are natural for studying quantum codes [15]. Recently, a link between quantum error correction and quantum gravity through the AdS/CFT duality has been explored [16], [17], [18], [19], [20], [21]. These holographic codes, and their tensor networks, have been applied in quantum computing, many-body quantum states, and quantum field theory [22], [23], [24], [25], [26], [27], [28], [29], [30].

In our previous work [31], we introduced the quantum lego formalism wherein we can construct large codes from small building blocks via tensor networks. There we were able to reconstruct several families of quantum codes such as surface codes and Bacon-Shor codes, as well as form new families of codes with various desirable properties, as sparse tensor networks. See also the related approach of [32]. Nonetheless, these methods do not predict when the resulting codes will be good error correction properties, and in particular one cannot predict the distance of the resulting code in general [33], [34]. In this work we aim to extend tensor network methods to quantum enumerators to address this problem.

Shor and Laflamme introduced two quantum weight enumerators [35] related by an analogue of the MacWilliams identity. For stabilizer codes, the A-enumerator counts the number of stabilizers at each Hamming weight (i.e. the number of non-identity Pauli operators) while the B-enumerator counts the number of logical operators at each Hamming weight. The distance of a stabilizer code is then the smallest degree where the coefficient (up to appropriate normalization) of B strictly exceeds that of A. In fact, the difference of B and A expresses the probability of an undetected error even when the underlying code is not a stabilizer code [36].

Other quantum enumerators have been introduced, such as the “shadow” enumerator of [37] and [38] to create bounds on the length, dimension, and distance for which quantum codes can exist. In [39] and [40], the authors introduce quantum enumerators that count the types of Pauli operators that appear in stabilizer/logical operators, for the purpose of analyzing asymmetric error models.

Before we can apply tensor networks to analyze weight enumerators, we need to construct a novel formalism that is compatible with both concepts. To this end, we first introduce the notion of a tensor enumerator, which would be the elementary building block of the tensor network for quantum weight enumerators. These are tensors in the usual sense of multi-linear algebra, however with coefficients that are polynomials. We then present our main result: if one has a large quantum code that is created by contracting a tensor network of small quantum codes, then the quantum weight enumerators of the large code can be obtained by contracting precisely the same network over tensor enumerators of the small codes. This allows us to break down the difficult problem of computing quantum weight enumerator polynomials first into smaller manageable pieces, which can be solved easily, then put back together.

Generally, the computation of a quantum weight enumerator, including a tensor enumerator, is exponential in the number of qubits. For small codes this is feasible. However, for large quantum codes created by the lego formalism the cost is exponential in the largest cut-distance encountered when contracting the network, which is typically far smaller than the number of qubits. We go into this fact in greater detail, and also explore other applications to quantum error correction, in a companion paper [41].

In this paper, we focus on the novel theoretical construction of tensor enumerators and prove their critical properties. Our contributions include (i) covariance under local unitary transformation (Theorem 27), (ii) tensor-product functorality (Proposition 29), (iii) a MacWilliams identity (Theorem 32), and (iv) invariance under the tensor network

0018-9448 © 2023 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

Authorized licensed use limited to the terms of the applicable license agreement with IEEE. Restrictions apply.
trace (Theorem 43). Along the way, we also provide a unified proof for all quantum MacWilliams identities in their most general form (Theorem 35).

II. Definitions

In this section we provide standard constructions of error bases and quantum enumerators from the literature. Our error basis will be a more restrictive case of the nice error bases of [42], which allows us to prove some critical results related to MacWilliams identities. Nonetheless, the quantum enumerators defined by Shor and Laflamme [35, as well as the so-called complete quantum enumerators [39, 43], can be defined over an arbitrary unitary bases of a Hilbert space. We also introduce the double quantum enumerators of [40], defined over the error basis of Pauli operators.

A unitary basis on a Hilbert space $\mathcal{H}$ of dimension $q$ is a set of unitary operators $E$ containing $I$ that form an orthonormal basis of $L(\mathcal{H})$, the space of linear operators on $\mathcal{H}$, in that $\text{Tr}(E^\dagger F) = 0$ if $E \neq F$ [44]. We focus on a class of “nice error bases” [42, 45], those with Abelian index group [43, 46]. To avoid unnecessary repetition, we will simply refer to these as an “error basis” formally defined as follows.

Definition 1: An error basis on a Hilbert space is a unitary basis $\mathcal{E}$ that additionally satisfies

$$ EF = \omega(E, F) FE \quad \text{(II.1)} $$

for all $E, F \in \mathcal{E}$, where $\omega(E, F) \in \mathbb{C}$ satisfies $\omega(E, F)^r = 1$ for some fixed $r$.

Perhaps the most important example of such a basis arises from the (generalized) Pauli group. This group is generated by operators $X, Z, \text{ and } H$.

$$ Z|j\rangle = e^{\pi i / q} |j\rangle \quad \text{and} \quad X|j\rangle = |j + 1 \bmod q\rangle, \quad \text{(II.2)} $$

where $\zeta = e^{2\pi i / q}$. That is, the eigenvalues of $Z$ are the $q$-th roots-of-unity, and on its eigenbasis $X$ acts as cyclic shift. Then $XZ = \zeta^{-1} ZX$. The Pauli basis is

$$ \mathcal{P} = \{ X^aZ^b : a, b \in \{ 0, \ldots, q - 1 \} \}, \quad \text{(II.3)} $$

and $\omega(X^aZ^b) = \zeta^{(a-b)(a-b)}$.

Given an error basis for $H$, we obtain one for $H^\otimes n$ via

$$ \mathcal{E}^n = \{ E_1 \otimes \cdots \otimes E_n : \text{ each } E_j \in \mathcal{E} \}. \quad \text{(II.4)} $$

Equation (II.1) holds where we define $\omega$ on $\mathcal{E} \times \mathcal{E}$ to be the product of the $\omega$ on each component. In particular, for the Pauli group $\omega(X^aZ^b, X^cZ^d) = e^{2\pi i (a-b)(c-d)}$, where $X^aZ^b = X^a \otimes \cdots \otimes X^a$ and similarly for the other terms.

The weight of such an operator is the number of non-identity tensor factors it contains, $\text{wt}(E_1 \otimes \cdots \otimes E_n) = |\{ r : E_r \neq I \}|$.

Write $\mathcal{E}(d)$ for the set of weight $d$ operators in the error basis.

Given an error basis $\mathcal{E}$ on a Hilbert space $\mathcal{H}$, Shor and Laflamme [35] define two quantum weight distributions\footnote{We opt for the normalization in [37] as opposed to that in [35], and take a different convention for which term is conjugate linear.} for a pair of Hermitian operators $M_1$ and $M_2$ on $\mathcal{E}(\otimes n)$,

$$ A_d(M_1, M_2) = \sum_{E \in \mathcal{E}(d)} \text{Tr}(E^\dagger M_1) \text{Tr}(EM_2), \quad \text{(II.5)} $$

$$ B_d(M_1, M_2) = \sum_{E \in \mathcal{E}(d)} \text{Tr}(E^\dagger M_1 E M_2). \quad \text{(II.6)} $$

We will often drop the notation $M_1, M_2$ from this notation when they are understood. While it is not obvious from this definition, both these values are invariant under local unitary transformations. That is, if $U = U_1 \otimes \cdots \otimes U_n$ is any local unitary then

$$ A_d(U M_1 U^\dagger, U M_2 U^\dagger) = A_d(M_1, M_2) \quad \text{(II.7)} $$

and similarly for $B_d$. For completeness, we provide a proof in §III but will show this in a broader context in later sections.

Definition 2: The weight enumerators $A$ and $B$ of Hermitian operators $M_1, M_2$ are

$$ A(z; M_1, M_2) = \sum_{d=0}^{n} A_d(M_1, M_2) z^d $$

$$ = \sum_{E \in \mathcal{E}^n} \text{Tr}(E^\dagger M_1) \text{Tr}(EM_2) z^{\text{wt}(E)} \quad \text{(II.8)} $$

$$ B(z; M_1, M_2) = \sum_{d=0}^{n} B_d(M_1, M_2) z^d $$

$$ = \sum_{E \in \mathcal{E}^n} \text{Tr}(E^\dagger M_1 EM_2) z^{\text{wt}(E)}. \quad \text{(II.9)} $$

As above, we will often drop the notation $M_1, M_2$ when the context is clear, or simplify it when $M_1 = M_2$ and refer to just the one operator. Additionally we will often find it useful to consider the enumerators in homogeneous form,

$$ A(w, z) = w^n A(z/w) $$

$$ = \sum_{d=0}^{n} A_d(M_1, M_2) w^{n-d} z^d, \quad \text{(II.10)} $$

$$ B(w, z) = w^n B(z/w) $$

$$ = \sum_{d=0}^{n} B_d(M_1, M_2) w^{n-d} z^d, \quad \text{(II.11)} $$

for which we do not introduce new notation. These two enumerators are not independent, being linked by the quantum MacWilliams identity [35, 37]. In the case of $\mathcal{H} = \mathbb{C}^q \otimes \mathbb{C}^q$ this reads:

$$ B(w, z) = A \left( \frac{w+(q^2-1)z}{q}, \frac{w-z}{q} \right). \quad \text{(II.12)} $$

We will provide a simple proof of this in §VI below, as a consequence of a more general result.

For classical codes with an alphabet size greater than two, a “complete” enumerator may be constructed that also has desirable properties such as a MacWilliams identity [47, 48]. Hu et al. [39] noted that quantum codes can be viewed in this light. For each $R \in \mathcal{E}$ introduce a weight function that counts the number of occurrences of $R$, $\text{wt}_R(E_1 \otimes \cdots \otimes E_n) = |\{ r : E_r = R \}|$ and an indeterminate $u_R$. The complete enumerators are

$$ E(\{u_R\}; M_1, M_2) = \sum_{G \in \mathcal{E}^n} \text{Tr}(G^\dagger M_1) \text{Tr}(GM_2) \prod_{R \in \mathcal{E}} u_R^{\text{wt}_R(G)} \quad \text{(II.13)} $$

$$ F(\{u_R\}; M_1, M_2) = \sum_{G \in \mathcal{E}^n} \text{Tr}(G^\dagger M_1 GM_2) \prod_{R \in \mathcal{E}} u_R^{\text{wt}_R(G)}. \quad \text{(II.14)} $$
These enumerators are also obtained by Li and Xing [43] through the group algebra associated to the error basis. In any case, these polynomials are homogeneous of degree $n$. They are related to the usual Shor-Laflamme enumerators by $A(w, z) = E(w, z, \ldots, z)$ and $B(w, z) = F(w, z, \ldots, z)$.

For the Pauli basis $P$ of local dimension $q$, Hu, Yang, and Yau define a double enumerator as follows [40]. Separate $X$ and $Z$ weights are defined by

$$\text{wt}^X(P) = |\{j : P_j = X^a Z^{a'} \text{ where } a' \neq 0\}|$$

(II.15)

$$\text{wt}^Z(P) = |\{j : P_j = X^a Z^{a'} \text{ where } a' \neq 0\}|$$

(II.16)

For example on $\mathbb{C}^2$, we have $\text{wt}^X(\sigma_x) = \text{wt}^X(\sigma_y) = 1$ and $\text{wt}^Z(\sigma_y) = \text{wt}^Z(\sigma_z) = 1$ (other values being zero). The double enumerators are defined as

$$C(x, z; M_1, M_2) = \sum_{P \in P^n} \text{Tr}(P^\dagger M_1 \text{Tr}(PM_2)) x^{\text{wt}^X(P)} z^{\text{wt}^Z(P)}$$

(II.17)

$$D(x, z; M_1, M_2) = \sum_{P \in P^n} \text{Tr}(P^\dagger M_1 PM_2) x^{\text{wt}^X(P)} z^{\text{wt}^Z(P)}.$$ 

(II.18)

To homogenize the double enumerators, we introduce two variables $w, y$ and define $C(w, x, y, z) = w^a y^a C(x/y, z/w)$ and likewise for $D$. That is, we homogenize in bidegree $(n, n)$ with $y$ as the homogenizing variable for $x$, and $w$ for $z$.

Both the double and complete enumerators are connected with versions of the quantum MacWilliams identity [40]. Like the MacWilliams identity for the Shor-Laflamme enumerators, these will follow from a more general approach given in §VI.

### III. Scalar Enumerators

Our goal is to extend tensor network methods to include enumerators. In this theory, our scalars are elements of polynomial rings, and hence we refer to the usual quantum weight enumerators as “scalar” enumerators. We will focus on the Shor-Laflamme enumerators, which in general are complex-valued polynomials in one variable $A, B \in \mathbb{C}[z]$, but in application (namely when $M_1 = M_2$) they have real-valued coefficients. Most of our results also apply the double enumerators, $C, D \in \mathbb{C}[x, z]$, and complete enumerators, $E, F \in \mathbb{C}[\{w_R\}]$. Much of the material in this section is well-known. However we do provide some novel results, centering on the enumerators of quantum states and the relationship between the enumerators of a stabilizer code and the Choi state of its encoding circuit.

Fix an error basis $E$ on a Hilbert space $\mathcal{H}$. For $E \in \mathcal{E}_n^+$ define the support of $E$, $\text{supp}(E)$, as the set of indices $j$ with $E_j \neq I$. In particular, the $\text{wt}(E) = |\text{supp}(E)|$. Let $U = U_1 \otimes \cdots \otimes U_n$ be a local unitary transformation on $\mathcal{H}^\otimes n$. Note $\text{supp}(U^\dagger EU) = \text{supp}(E)$ for any $E \in \mathcal{E}_n$. As the elements of $\mathcal{E}_n^+$ form an orthogonal basis (under trace-norm) for $\mathcal{H}^\otimes n$, we can write $U^\dagger EU = \frac{1}{q^n} \sum_{F \in \mathcal{E}_n} \text{Tr}(U^\dagger EU F^\dagger) F$. Note that unless $\text{supp}(F) = \text{supp}(U^\dagger EU) = \text{supp}(E)$ then at least one factor in $U^\dagger EU F^\dagger$ is not the identity and hence $\text{Tr}(U^\dagger EU F^\dagger) = 0$. Therefore,

$$U^\dagger EU = \frac{1}{q^n} \sum_{F : \text{supp}(F) = \text{supp}(E)} \text{Tr}(U^\dagger EU F^\dagger) F.$$ 

(III.1)

An identical argument applies for $E^\dagger$.

**Lemma 3:** We have

$$\frac{1}{q^n} \sum_{F : \text{supp}(F) = \text{supp}(E)} \text{Tr}(U^\dagger E^\dagger UF)^\dagger \text{Tr}(U^\dagger E^\dagger UF)$$

$$= \begin{cases} q^n & \text{if } F = G \in \mathcal{E}_n^+[d] \\ 0 & \text{otherwise}. \end{cases}$$

(III.2)

**Proof:** Using (III.1), and its analogue for $E^\dagger$, we find for any $E \in \mathcal{E}_n^+[d]$ that

$$q^n = \text{Tr}(I) = \text{Tr}(U^\dagger E^\dagger UF)$$

$$= \frac{1}{q^n} \sum_{F, G \in \mathcal{E}_n^+[d]} \text{Tr}(U^\dagger E^\dagger UF) \text{Tr}(U^\dagger E^\dagger UF^\dagger) \text{Tr}(F^\dagger G)$$

$$= \frac{1}{q^n} \sum_{F \in \mathcal{E}_n^+[d]} \text{Tr}(U^\dagger E^\dagger UF) \text{Tr}(U^\dagger E^\dagger UF^\dagger).$$

(III.3)

Exchanging the role of $E$ and $F$ in this final expression proves the claimed result with $F = G \in \mathcal{E}_n^+[d]$.

In the case $F = G$, but they are not in $\mathcal{E}_n^+[d]$, then we have already seen $\text{Tr}(U^\dagger EU F^\dagger) = 0$ whenever $E \in \mathcal{E}_n^+[d]$.

Finally, if $F \neq G$ then similar to the above

$$0 = \text{Tr}(FG^\dagger) = \text{Tr}(UFU^\dagger G^\dagger)$$

$$= \frac{1}{q^n} \sum_{E_1, E_2 \in \mathcal{E}_n^+[d]} \text{Tr}(UFU^\dagger E_1^\dagger) \text{Tr}(UG^\dagger U^\dagger E_2) \text{Tr}(E_1 E_2^\dagger)$$

$$= \frac{1}{q^n} \sum_{E \in \mathcal{E}_n^+[d]} \text{Tr}(UFU^\dagger E^\dagger) \text{Tr}(UG^\dagger U^\dagger E)$$

$$= \frac{1}{q^n} \sum_{E \in \mathcal{E}_n^+[d]} \text{Tr}(U^\dagger E^\dagger UF) \text{Tr}(U^\dagger E^\dagger UF^\dagger).$$

(III.4)

as claimed.

**Theorem 4:** Let $U$ be any local unitary on $\mathcal{H}_n^\otimes$. Then

$$A(\cdot; UM_1 U^\dagger, UM_2 U^\dagger) = A(\cdot; M_1, M_2).$$

(III.5)

**Proof:** We expand $M_1 = \frac{1}{q^n} \sum_F \text{Tr}(F^\dagger M_1) F$ and similarly $M_2 = \frac{1}{q^n} \sum_G \text{Tr}(GM_2^\dagger) G^\dagger$. Then from the lemma,

$$A(\cdot; UM_1 U^\dagger, UM_2 U^\dagger)$$

$$= \frac{n}{q^n} \sum_{d=0}^n \sum_{E \in \mathcal{E}_n^+[d]} \text{Tr}(U^\dagger E^\dagger UM_1) \text{Tr}(U^\dagger E^\dagger UM_2) z^d$$

$$= \frac{n}{q^n} \sum_{d=0}^n \sum_{F, G \in \mathcal{E}_n^+[d]} \sum_{E, E' \in \mathcal{E}_n^+[d]} \text{Tr}(U^\dagger E^\dagger UF) \text{Tr}(U^\dagger E^\dagger UF^\dagger) \text{Tr}(F^\dagger M_1) \text{Tr}(GM_2^\dagger) z^d$$

$$= \frac{n}{q^n} \sum_{d=0}^n \sum_{F \in \mathcal{E}_n^+[d]} \text{Tr}(F^\dagger M_1) \text{Tr}(FM_2^\dagger) z^d$$

$$= A(\cdot; M_1, M_2).$$

(III.6)
One could provide a similar proof for the invariance of $B$, however this follows immediately from the quantum MacWilliams identity:

\[
B(w, z; U M_1 U^\dagger, U M_2 U^\dagger) = \mathcal{A}\left(\frac{w+\sqrt{q-1}z}{q}, \frac{w-z}{q}; U M_1 U^\dagger, U M_2 U^\dagger\right)
= \mathcal{A}\left(\frac{w+\sqrt{q-1}z}{q}, \frac{w-z}{q}; M_1, M_2\right)
= B(w, z; M_1, M_2).
\]

(III.7)

Note that none of C, D, E, or F are locally unitarily invariant as they depend on counting specific Pauli operators in each factor.

**Lemma 5.** For Hermitian operators $M_1, M_2$ on Hilbert space $\mathcal{S}$ and $M'_1, M'_2$ on $\mathcal{S}'$, we have

\[
\mathcal{A}(z; M_1 \otimes M'_1, M_2 \otimes M'_2) = \mathcal{A}(z; M_1, M_2) \cdot \mathcal{A}(z; M'_1, M'_2)
\]

(III.8)

and similarly for B, C, D, E, F.

**Proof:** Note that weights are additive in the sense that if $wt(E_1) = d_1$ and $wt(E_2) = d_2$ then $wt(E_1 \otimes E_2) = d_1 + d_2$. Then

\[
\mathcal{A}(z; M_1, M_2) \cdot \mathcal{A}(z; M'_1, M'_2)
= \sum_{E_1} \mathcal{Tr}(E_1^\dagger M_1) \mathcal{Tr}(E_1 M'_1) z^{wt(E_1)}
\cdot \sum_{E_2} \mathcal{Tr}(E_2^\dagger M_2) \mathcal{Tr}(E_2 M'_2) z^{wt(E_2)}
= \sum_{E_1, E_2} \mathcal{Tr}((E_1 \otimes E_2)^\dagger M_1 M_2) \cdot \mathcal{Tr}((E_1 \otimes E_2) M'_1 M'_2) z^{wt(E_1) + wt(E_2)}
= \mathcal{A}(z; M_1 \otimes M'_1, M_2 \otimes M'_2).
\]

(III.9)

The proofs for B, C, D, E, F are similar and so omitted. □

**Example 6 (35):** Let $\Pi_\mathcal{C}$ be the projection onto $\{0, 1\}^n$ binary stabilizer code $\mathcal{C}$, and write $\mathcal{S} = \mathcal{S}(\mathcal{C}) = \{S_1, \ldots, S_{n-k}\}$ for its stabilizer group. Then

\[
\Pi_\mathcal{C} = \frac{1}{2^{n-k}} \sum_{j=1}^{n-k} (I + S_j) = \frac{1}{2^{n-k}} \sum_{S \in \mathcal{S}} S.
\]

(III.10)

For any Pauli operator $P \in \mathcal{P}^n$ we have

\[
\sum_{S \in \mathcal{S}} \mathcal{Tr}(PS) = \begin{cases} 2^n & \text{if } P \in \mathcal{S}, \\ 0 & \text{otherwise}. \end{cases}
\]

(III.11)

Therefore

\[
A_d = \frac{1}{4^n-k} \sum_{P \in \mathcal{P}^n[d]} \sum_{S, S' \in \mathcal{S}} \mathcal{Tr}(PS) \mathcal{Tr}(PS')
= 4^k \cdot |\mathcal{P}^n[d]| \cap \mathcal{S}.
\]

(III.12)

That is, properly normalized, the weight enumerator

\[
\frac{1}{4^k} \cdot \mathcal{A}(z; \Pi_\mathcal{C}) = \sum_{d=0}^{2^n} |\mathcal{P}^n[d] \cap \mathcal{S}| z^d
\]

counts the number of stabilizers of each weight.

If a Pauli operator $P \in \mathcal{N} = \mathcal{N}(\mathcal{C})$ (the normalizer of the code) then $\Pi_\mathcal{C} = \Pi_\mathcal{C} P$ and so

\[
\mathcal{Tr}(\Pi_\mathcal{C} P \Pi_\mathcal{C}) = \mathcal{Tr}(\Pi_\mathcal{C}) = 2^k.
\]

(III.14)

On the other hand, if $P \notin \mathcal{N}$ then there is some $j_0$ where $PS_{j_0} = -S_{j_0} P$. Thus

\[
\Pi_\mathcal{C} P \Pi_\mathcal{C} \propto (I - S_{j_0})(I + S_{j_0}) \prod_{j \neq j_0} (I \pm S_j)(I + S_j) = 0.
\]

(III.15)

Therefore

\[
B_d = \sum_{P \in \mathcal{P}^n[d]} \mathcal{Tr}(\Pi_\mathcal{C} P \Pi_\mathcal{C}) = 2^k \cdot |\mathcal{P}^n[d] \cap \mathcal{N}|,
\]

(III.16)

and

\[
\frac{1}{2^k} B(z; \Pi_\mathcal{C}) = \sum_{d=0}^{2^n} |\mathcal{P}^n[d] \cap \mathcal{N}| z^d
\]

(III.17)

counts the number of normalizers of each weight. Hence, the distance of the code $\mathcal{C}$ is the smallest $d$ such that $A_d \neq B_d$.

Note that the above example carries over to $q$-ary stabilizer codes [49], adjusting the normalization appropriately. The double and complete weight enumerators further refine this information allowing separate analysis of $X$ and $Z$ errors [50], [51], [52].

**Lemma 7:** Let $|\psi\rangle$ be any state. Then its projector has

\[
\mathcal{A}(z; |\psi\rangle \langle \psi|) = \frac{1}{2^k} \sum_{d=0}^{2^n} |\mathcal{P}^n[d] \cap \mathcal{N}| z^d
\]

(III.18)

**Definition 8:** Let $\mathcal{C}$ be a $[[n, k]]$ quantum code. Its encoding tensor is the (unnormalized) Choi state $|T_\mathcal{C}\rangle = \sum_{x \in \mathcal{S}_2} |x_L\rangle \otimes |x\rangle = \sum_{x \in \mathcal{S}_2} U(|x\rangle \otimes |0\rangle^{(n-k)}) \otimes |x\rangle$,

(III.19)

where $U$ is the unitary encoding map of the code.

**Theorem 9:** Let $\mathcal{C}$ be a $[[n, 1]]$-stabilizer code, $|T_\mathcal{C}\rangle$ its encoding tensor, and $\Pi_\mathcal{C}$ the projection onto $\mathcal{C}$. Then

\[
A_d(|T_\mathcal{C}\rangle \langle T_\mathcal{C}|) = \frac{1}{4} A_d(\Pi_\mathcal{C}) + \frac{1}{2} B_{d-1}(\Pi_\mathcal{C}) - \frac{1}{4} A_{d-1}(\Pi_\mathcal{C}).
\]

(III.20)

In particular, the distance of the code is the smallest $d$ such that $A_{d+1}(|T_\mathcal{C}\rangle \langle T_\mathcal{C}|) > \frac{1}{2} A_{d+1}(\Pi_\mathcal{C})$.

**Proof:** First note that the normalizer of the code is the disjoint union

\[
\mathcal{N}(\mathcal{C}) = \mathcal{S}(\mathcal{C}) \cup X \mathcal{S}(\mathcal{C}) \cup Y \mathcal{S}(\mathcal{C}) \cup Z \mathcal{S}(\mathcal{C})
\]

(III.21)

where $X$, $Y$, $Z$ are any choice of representations for the logical Pauli operators. Recall that $\frac{1}{4} B_d(\Pi_\mathcal{C}) = |\mathcal{N}(\mathcal{C}) \cap \mathcal{P}^n[d]|$ and $\frac{1}{4} A_d(\Pi_\mathcal{C}) = |\mathcal{S}(\mathcal{C}) \cap \mathcal{P}^n[d]|$, and so the number of proper logical operators of weight $d$ is $\frac{1}{2} B_d(\Pi_\mathcal{C}) - \frac{1}{4} A_d(\Pi_\mathcal{C})$. 

Authorized licensed use limited to the terms of the applicable license agreement with IEEE. Restrictions apply.
The encoding tensor state \( |T_ε \rangle \) is a stabilizer state (namely a \([n+1,0]\)-stabilizer code). If we write the unencoded qubit as the \((n+1)^{th}\) qubit of the state, the stabilizers of this state form the disjoint union

\[
S(|T_ε \rangle) = (S(ε) \otimes I) \cup (\overline{X}S(ε) \otimes X) \\
(\overline{Y}S(ε) \otimes Y) \cup (ZS(ε) \otimes Z)
\]  

where we have abused notation slightly to write, say, \( \overline{X}S(ε) \otimes X \). The weight elements of \( S(ε) \otimes I \) are naturally bijective with those of \( S(ε) \), of which there are \( \frac{1}{d} A_d (ε) \) of them. The weight \( d \) elements of \( \overline{X}S(ε) \otimes X \) are naturally bijective with the weight \( d-1 \) elements of \( \overline{X}S(ε) \), similarly for \( \overline{Y} \) and \( \overline{Z} \). Hence there are \( \frac{1}{d} B_{d-1} (I_ε) = \frac{1}{d} A_{d-1} (I_ε) \) of these in total.

To illustrate this theorem, consider the perfect \([5,1,3]\) code and its encoding tensor, which is the perfect \([6,0,4]\) tensor, that play a prominent role in the constructions of \([16]\). For ease of reading, let us simply write \( A_{[5,1,3]} \), etc and similar for other codes. We find

\[
\frac{1}{4} A_{[5,1,3]} (z) = 1 + 15 z^4, \\
\frac{1}{4} B_{[5,1,3]} (z) = 1 + 30 z^3 + 15 z^4 + 18 z^5, \\
A_{[6,0,4]} (z) = 1 + 45 z^4 + 18 z^6.
\]

Clearly this theorem cannot hold for \( k > 1 \). In fact, one would not expect any natural extension of this result to \( k > 1 \) using the scalar enumerators we have discussed above. Consider for instance a quantum code \( C \), with \( k = 2 \); the encoding tensor \( |T_ε \rangle \) will be stabilized by sets of operators \((X \otimes I)S(ε) \otimes X \otimes I \) and \((\overline{X} \otimes Y)S(ε) \otimes X \otimes Y \), among others. In weight \( d \), operators in the first set are counted in \( B_d (I_ε) \) and \( A_{d-1} (|T_ε \rangle \langle T_ε |) \) while those in the second set are also counted in \( B_d (I_ε) \), but instead in \( A_{d+2} (|T_ε \rangle \langle T_ε |) \). Hence to obtain such a result as in the theorem, we would need to know how to decompose \( B_d (I_ε) \) into counts of logical operators based on the weight of the logical operator they represent. To illustrate this point, consider the \( 2 \times 2 \) Bacon-Shor codes as a \([4,2,2]\) stabilizer code, and it encoding tensor as a \([6,0,3]\) stabilizer state, that play a prominent role in the constructions of \([31]\). Extending our shorthand notation to this case

\[
\frac{1}{16} A_{[4,2,2]} (z) = 1 + 3 z^4, \\
\frac{1}{4} B_{[4,2,2]} (z) = 1 + 18 z^3 + 24 z^4 + 21 z^5, \\
A_{[6,0,3]} (z) = 1 + 8 z^3 + 21 z^4 + 24 z^5 + 10 z^6.
\]

Even allowing for changing the normalization terms in \( A \) and \( B \), these would not satisfy \((3.20)\).

Working in generality, if \( |ψ⟩ \) is any quantum state then by Lemma 7 we have \( A(w,z; |ψ⟩⟨ψ|) = B(w,z; |ψ⟩⟨ψ|) \) and hence this enumerator is invariant under the quantum MacWilliams transform, akin to the situation of classical self-dual codes. We can write the quantum MacWilliams transform as a linear transformation of the variables

\[
\begin{pmatrix} w \\ z \end{pmatrix} \rightarrow \frac{1}{q} \begin{pmatrix} 1 & q^2 - 1 \\ 1 & -1 \end{pmatrix} \begin{pmatrix} w \\ z \end{pmatrix}.
\]

Thus enumerator polynomials of states \( A(w,z; |ψ⟩⟨ψ|) \) must be left invariant by this transformation, and in fact the whole group

\[
G = \left\{ \frac{1}{q} \begin{pmatrix} 1 & q^2 - 1 \\ 1 & -1 \end{pmatrix} \right\}.
\]

The collection of polynomials left invariant by linear group action on its variables is called the invariant ring of the group; there is much literature dedicated to their study, see for instance \([53, 54]\). In particular, since our case \( |G| = 2 \) this ring is trivial to compute and one finds

\[
A(w,z; |ψ⟩⟨ψ|) \in \mathbb{R}[w + (q - 1)z, w^2 + (q - 2)wz + (q^2 - q + 1)z^2]
\]

belongs to the above set of real polynomials.

In the case of \( q = 2 \), we can make this more explicit as

\[
A(w,z; |ψ⟩⟨ψ|) = w + z,
\]

where \( |β⟩ = \frac{1}{\sqrt{2}}(|00⟩ + |11⟩) \) is the Bell state. That is, the enumerator polynomial of any state lies in the ring is freely generated by the enumerators of a one-qubit state and a maximally entangled two-qubit state.

Still working with \( q = 2 \), we refine this somewhat for codes whose encoding tensors are “even” in the sense that all their stabilizers have even Hamming weight, such as the Bell state or the \([6,0,4]\) tensor as shown above. Then, the enumerator lies in the invariant ring of

\[
\frac{1}{2} \begin{pmatrix} 1 & 3 \\ 1 & -1 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix},
\]

which is isomorphic to the dihedral group \( D_6 \). Again this has small order, so the invariant ring is easily computed using the algebraic software \( SAGE \) \([55]\), yielding that for any projector onto any even encoding tensor

\[
A(w,z; |T_ε \rangle \langle T_ε |) \in \mathbb{R}[w^2 + 3z^2, w^6 + 15w^4 z^2 + 15w^2 z^4 + 33z^6]
\]

\[
= \mathbb{R}[A(w,z; |β⟩⟨β|), A(w,z; |T_θ⟩⟨T_θ |)]
\]

where \( |T_θ⟩ \) is the encoding tensor of the code \( θ \) with stabilizer group \( S(θ) = \langle YYIII, YIYII, YIIYI, YIIIY \rangle \).

IV. VECTOR ENUMERATORS

While we are ultimately interested in tensor enumerators, we find it illuminating to first start with the case of vector enumerators. We call this class of enumerators “vectors” as they are single index tensors and so play the analogous role of vectors in multilinear algebra. Nonetheless, we will see that they also behave like matrices, in that local unitary action transforms the a vector enumerator via an adjoint representation, and that vector enumerators support a natural trace operation to scalar enumerators. We will also state a quantum MacWilliams identity for vector enumerators, however postpone its proof for a later section.

Formally, let \( E \) be an error basis on \( S_3 \), and as before write \( E^{n-1} = \{ E_1 \otimes \cdots \otimes E_{n-1} : E_1, \ldots, E_{n-1} \in E \} \).
and $\mathcal{E}^{-1}[d] = \{E \in \mathcal{E}^{-1} : \text{wt}(E) = d\}$. Fix a “leg” $j \in \{1, \ldots, n\}$ and for $E \in \mathcal{E}$ and $F \in \mathcal{E}^{-1}$ define
$$E \otimes_j F = F_1 \otimes \cdots \otimes F_{j-1} \otimes E \otimes F_j \otimes \cdots \otimes F_n \in \mathcal{E}^n.$$  
(IV.1)

That is, $E \otimes_j F$ has $E$ inserted as the $j$-th factor into $F$. For a pair of elements of this basis $(E, E')$ define weights of the Hermitian operators $M_1, M_2$ on $\mathfrak{U} \mathfrak{H}^\otimes n$ as
$$A_d^{(j)}(E, E'; M_1, M_2) = \sum_{F \in \mathcal{E}^{-1}[d]} \text{Tr}((E \otimes_j F)^{M_1}) \text{Tr}((E' \otimes_j F)^{M_2}),$$  
(IV.2)

$$B_d^{(j)}(E, E'; M_1, M_2) = \sum_{F \in \mathcal{E}^{-1}[d]} \text{Tr}((E \otimes_j F)^{M_1}(E' \otimes_j F)^{M_2}).$$  
(IV.3)

These weight distributions are related to the usual $A$-weight (II.5) and $B$-weight (II.6) of $M_1$ and $M_2$, refined by specifying which error operators $E, E'$ appear in factor $j$. Note that the weights of the operators $E$ and $E'$ do not contribute to $A_d^{(j)}$ or $B_d^{(j)}$.

We formally define vector enumerators as the enumerators of these weights. However, these weights are indexed by a pair $(E, E')$, and so vector enumerators also have the characteristics of a matrix. As a matrix, they are symmetric when $M_1 = M_2$ (and symmetric in case of usual qubit Pauli basis $\mathcal{P}^n$). Formally, let us write $e_{E,E'}$ for the matrix unit, which is 1 in $(E, E')$ position and zero elsewhere. Denote $V = \text{span}_C\{e_{E,E'} : E, E' \in \mathcal{E}\}$ for this matrix space.

Definition 10: Let $\mathcal{E}$ be an error basis of Hilbert space $\mathfrak{U}$. The vector enumerators for Hermitian operators $M_1, M_2$ on $\mathfrak{U} \mathfrak{H} \mathfrak{H}^\otimes n$ at “leg” $j$ are
$$A^{(j)}(z; M_1, M_2) = \sum_{E,E' \in \mathcal{E}} \sum_{d=0}^n A_d^{(j)}(E, E'; M_1, M_2) z^d e_{E,E'},$$  
(IV.4)

$$B^{(j)}(z; M_1, M_2) = \sum_{E,E' \in \mathcal{E}} \sum_{d=0}^n B_d^{(j)}(E, E'; M_1, M_2) z^d e_{E,E'},$$  
(IV.5)

where $A_d^{(j)}$ and $B_d^{(j)}$ are defined in (IV.2) and (IV.3).

Our enumerators $A^{(j)}(z)$ and $B^{(j)}(z)$ are elements of the formal algebraic tensor product $C[z] \otimes_C V$, which is to say they are vectors in $V$ whose coefficients are complex-valued polynomials in $z$. We can define a “trace” operation on vectors in $V$ by summing the diagonal $(E, E)$, with an appropriate weight, as follows.

Definition 11: Define $\overline{\text{tr}} : C[z] \otimes_C V \to C[z]$ by taking the linear extension of
$$\overline{\text{tr}}(e_{E,E'}) = \begin{cases} 1 & \text{if } E = E' = I, \\ z & \text{if } E = E' \neq I, \\ 0 & \text{if } E \neq E'. \end{cases}$$  
(IV.6)

We have introduced the weight into this trace operation so that the trace of a vector enumerator of two Hermitian operators results in the corresponding scalar enumerator of those operators. While perhaps not overly useful at this stage, when we extending this operation to tensor enumerators in the next section, it will allow us to recover tensor enumerators of lower degree, which is an essential step in application to tensor networks. The proof the following result is clear and so left to the reader.

Lemma 12: For each leg $j$, we have
$$\text{tr}(A^{(j)}(z; M_1, M_2)) = A(z; M_1, M_2)$$
$$\text{tr}(B^{(j)}(z; M_1, M_2)) = B(z; M_1, M_2).$$

Note that vector enumerators are not invariant under local unitary operations. Rather, they are covariant with respect to an adjoint, or “Bloch sphere” style, representation of unitaries on $V$. Specifically, let $U$ be a unitary on $\mathfrak{U}$ and define $c_{GE} = \frac{1}{q} \text{Tr}(U^\dagger E U G^\dagger)$ from which we have
$$U^\dagger E U = \sum_G G c_{GE}.$$  
(IV.7)

Taking the Hermitian conjugate of (IV.7) we obtain
$$U^\dagger E^\dagger U = \sum_G G^\dagger c_{G^\dagger E^\dagger}.$$  
(IV.8)

Switching the roles of $E$ and $G$ in the definition of $c_{EF}$ shows
$$U E U^\dagger = \sum_G c_{E^\dagger G^\dagger},$$  
(IV.9)

and taking the Hermitean conjugate of this yields
$$U E^\dagger U^\dagger = \sum_G c_{E G^\dagger}.$$  
(IV.10)

The coefficients $c_{EF}$ then define a linear transformation on $V$, $\Lambda(U) \in L(V)$, as follows.

Definition 13: Let $\mathcal{E}$ be an error group on a Hilbert space $\mathfrak{U}$, and $V = \text{span}_C\{e_{E,E'} : E, E' \in \mathcal{E}\}$. Define $\Lambda : U(\mathfrak{U}) \to L(V)$ by
$$\Lambda(U)(e_{E,E'}) = \sum_{G,G' \in \mathcal{E}} c_{E G^\dagger} e_{E^\dagger G^\dagger} e_{G' G'},$$  
(IV.11)

where $U(\mathfrak{U})$ is the set of unitaries supported on $\mathfrak{U}$. We extended $\Lambda(U)$ to all of $C[z] \otimes_C V$ linearly, for which we obtain the following transformation rule.

Theorem 14: Let $U = U_1 \otimes \cdots \otimes U_n$ be a local unitary operator. Then
$$A^{(j)}(z; U M_1 U^\dagger, U M_2 U^\dagger) = \Lambda(U_j) \left(A^{(j)}(z; M_1, M_2)\right).$$  
(IV.12)

This result is a special case of Theorem 27 below and so we do not provide a proof here.

Lemma 5 above showed scalar enumerators are multiplicative under tensor product. Vector enumerators have a similar property, however one only of the multiplicands will be a vector enumerator (the one that supported the leg of the enumerator). The other multiplicand will be a scalar enumerator, and the product is the usual scalar-vector product, with polynomials as coefficients. We state this formally as follows. Again, the proof of this result follows immediately from the general case of tensor enumerators, Proposition 29 below, and so is omitted.
Proposition 15: Let $M_1 \otimes N_1$ and $M_2 \otimes N_2$ be Hermitian operators on $\mathcal{H}_1 \otimes \mathcal{H}_2$ where $\mathcal{H}_1 = \bigotimes_{j=1}^{n_1} \mathcal{H}_j$ and $\mathcal{H}_2 = \bigotimes_{k=1}^{n_2} \mathcal{H}_k$. Then for $j = 1, \ldots, n$ we have
\[
A^{(j)}(z; M_1 \otimes N_1, M_2 \otimes N_2) = A^{(j)}(z; M_1, M_2) A(z; N_1, N_2) .
\]
and for $j = n+1, \ldots, n+m$ we have
\[
A^{(j)}(z; M_1 \otimes N_1, M_2 \otimes N_2) = A(z; M_1, M_2) A^{(j-n)}(z; N_1, N_2) .
\]

Analogous formulas hold for $B^{(j)}$.

When $M_1 = M_2 = \Pi_\mathcal{E}$ are the projection onto a binary stabilizer code $\mathcal{E}$, the vector enumerators carry refined information about the structure of the stabilizer and normalizer groups of $\mathcal{E}$. While vector enumerators are matrices, as long as the underlying code has distance $d \geq 2$, these matrices are diagonal except in a degenerate case (see Theorem 18 below).

Example 16: Consider the vector enumerator of the logical leg in an encoding of a $[[n, 1]]$ binary stabilizer code $\mathcal{E}$. Recall from the proof of Theorem 9 above that if $|T_\mathcal{E}\rangle$ is the encoding state, then the stabilizer of this state is the disjoint union
\[
\mathcal{S}(|T_\mathcal{E}\rangle) = (\mathcal{S}(\mathcal{E}) \otimes I) \cup (\bar{X} \mathcal{S} \mathcal{E} \otimes X) \cup (-\bar{Y} \mathcal{S} \mathcal{E} \otimes Y) \cup (\bar{Z} \mathcal{S} \mathcal{E} \otimes Z)
\]
and the projection onto this state is
\[
|T_\mathcal{E}\rangle\langle T_\mathcal{E}| = \frac{1}{2^{n-1}} \sum_{S \in \mathcal{S}(\mathcal{E})} S \otimes I + \bar{X} S \otimes X - \bar{Y} S \otimes Y + \bar{Z} S \otimes Z .
\]

Now, for any $P = I, X, Y, Z$ we have
\[
\text{Tr}((F \otimes P)|T_\mathcal{E}\rangle\langle T_\mathcal{E}|) = \begin{cases} 1 & \text{if } F \in \bar{P} \mathcal{S} \mathcal{E} \text{,} \\ 0 & \text{otherwise.} \end{cases}
\]

Therefore
\[
A^{(n+1)}(z; |T_\mathcal{E}\rangle\langle T_\mathcal{E}|) = \sum_{P=I,X,Y,Z} \sum_{j=0}^{n-1} |\bar{P} \mathcal{S} \mathcal{E} \cap \mathcal{E}^n[d]| z^d e_{P,P} .
\]

In particular, for the projection onto the code space $\Pi_\mathcal{E}$ we have
\[
A(\Pi_\mathcal{E}) = A^{(n+1)}(I, I; |T_\mathcal{E}\rangle\langle T_\mathcal{E}|) \text{ and } B(\Pi_\mathcal{E}) = \sum_{P=I,X,Y,Z} A^{(n+1)}(P, P; |T_\mathcal{E}\rangle\langle T_\mathcal{E}|) .
\]

Lemma 17: Let $\mathcal{E}$ be a $[[n, k]]$ stabilizer code. Then for $P \neq P' \in \mathcal{P}$, and any leg $j$ we have
1) $A^{(j)}(P, P')$ is nonzero if and only if $PP' \otimes_j I^{\otimes(n-1)}$ is a stabilizer;
2) $B^{(j)}(P, P')$ is nonzero if and only if $A^{(j)}(P, P')$ is nonzero.

Proof: Recall that if $\Pi_\mathcal{E} = \frac{1}{2^n} \sum_{S \in \mathcal{S}(\mathcal{E})} S$ is the projector onto the code space, then $\text{Tr}((P \otimes_j Q)|\Pi_\mathcal{E}\rangle\langle \Pi_\mathcal{E}|) \neq 0$ if and only if $P \otimes_j Q \in \mathcal{S}(\mathcal{E})$. Now $A^{(j)}(P, P')$ is nonzero if and only if there exists some $Q \in \mathcal{P}^{n-1}$ such that
\[
\text{Tr}((P \otimes_j Q)|\Pi_\mathcal{E}(P' \otimes_j Q)|\Pi_\mathcal{E}) \neq 0 ,
\]
and hence both $(P \otimes_j Q), (P' \otimes_j Q) \in \mathcal{S}(\mathcal{E})$ and so is their product $PP' \otimes_j I^{\otimes(n-1)}$.

Now $B^{(j)}(P, P') \neq 0$ if and only if there is a $Q$ such that
\[
\text{Tr}((P \otimes_j Q)|\Pi_\mathcal{E}(P' \otimes_j Q)|\Pi_\mathcal{E}) \neq 0 ,
\]
which happens if and only if $P \otimes_j Q \in \mathcal{N}(\mathcal{E})$, and
\[
(P \otimes_j Q)(P' \otimes_j Q) \in \mathcal{P}^{n-1} \otimes_j I^{\otimes(n-1)} \in \mathcal{S}(\mathcal{E}) .
\]

Theorem 18: Let $\mathcal{E}$ be a $[[n, k]]$ stabilizer code of distance $d \geq 2$. Then every for every $j = 1, \ldots, n$ we have either:
1) there is a Pauli operator $P$ and code $\mathcal{E}'$ such that
\[
A^{(j)}(\Pi_\mathcal{E}) = A(\Pi_{\mathcal{E}'}) \cdot (e_{I,I} + e_{I,P} + e_{P,I} + e_{P,P}) ,
\]
2) or $A^{(j)}(\Pi_\mathcal{E})$ and $B^{(j)}(\Pi_\mathcal{E})$ are diagonal.

Proof: From the lemma, if $A^{(j)}(\Pi_\mathcal{E})$ has off-diagonal terms then $P \otimes_j I^{\otimes(n-1)} \in \mathcal{S}(\mathcal{E})$ for some $P$. This implies $\mathcal{E} = |\psi\rangle \otimes \mathcal{E}'$ where $|\psi\rangle$ is the +1-eigenstate of $P$ and $\mathcal{E}'$ is a $(n-1)$-qubit stabilizer code. By Proposition 15 we have $A^{(j)}(\Pi_\mathcal{E}) = A^{(j)}(|\psi\rangle\langle \psi|) A(\Pi_{\mathcal{E}'})$ and $B^{(j)}(\Pi_\mathcal{E}) = B^{(j)}(|\psi\rangle\langle \psi|) B(\Pi_{\mathcal{E}'})$. Finally it is straightforward to show that this $|\psi\rangle$ we have
\[
A^{(j)}(|\psi\rangle\langle \psi|) = B^{(j)}(|\psi\rangle\langle \psi|) = e_{I,I} + e_{I,P} + e_{P,I} + e_{P,P} .
\]

Recall the “cleaning lemma” [56, Lemma 1] relates the supports of logical Pauli operators on a stabilizer code. It states: for any subset of qubits $J$ either there exists a nontrivial logical Pauli operators supported on $J$, or every logical Pauli operator has a representation is that is trivial on $J$ (that is it can be “cleaned” from $J$). In particular, if $|J| < d$, the distance of the code, then no nontrivial logical operator can be supported on $J$. Thus any subset of qubits of size less than the distance of the code can be cleaned. For our purposes, we are interested in a slightly different notion of cleaning. We suppose that every Pauli operator on $J$ extends to a stabilizer of the code. As there are $4^{|J|}$ Pauli operators on $J$ but only $2^{n-k}$ stabilizers, this can only hold when $|J| \leq \frac{1}{2}(n-k)$. When this condition is satisfied, we can clean any Pauli operator from $J$: given any Pauli operator $P$, we take the Pauli operator on $J$ given by $P|J\rangle$ and extend this to a stabilizer $S$. Then $PS$ is equivalent to $P$ and trivial on $J$.

Corollary 19: Let $\mathcal{E}$ be a stabilizer code and $|T_\mathcal{E}\rangle$ its encoding tensor. Then for any logical qubit $j$ we have $A^{(j)}(|T_\mathcal{E}\rangle\langle T_\mathcal{E}|)$ is diagonal.
Proof: By construction, along every logical leg $j$ every local Pauli operator gives rise to a stabilizer of $|T_E\rangle$, and thus logical legs can be cleaned in our sense above. Moreover, these stabilizers are not of the form $P \otimes T^{(n-1)}$ and therefore in Theorem 18 we are in the case of $A^{(j)}(|T_E\rangle<T_E>)$ being diagonal.

Definition 20: The reduced enumerator $\tilde{A}^{(j)}$ is the diagonal of $A^{(j)}$. Namely, if $A^{(j)} = \sum_{E,E'} \sum_{d=0}^{n-1} A^{(j)}_d(E,E')z^d e_{E,E'}$ then

$$\tilde{A}^{(j)} = \sum_{E} \sum_{d=0}^{n-1} A^{(j)}_d(E)z^d e_{E,E}. \quad (IV.27)$$

Identically, $\tilde{B}^{(j)}$ is the restriction of $B^{(j)}$ to the diagonal.

The content of Theorem 18 is that if $d \geq 2$ then, except in a trivial case, $A^{(j)} = \tilde{A}^{(j)}$ (and similarly $B^{(j)} = \tilde{B}^{(j)}$) for each $j = 1, \ldots, n$. To simplify the notation, we will write $e_E = e_{E,E}$ when working with reduced enumerators.

Next we present the analogue of the quantum MacWilliams identity for vector enumerators. This theorem is a special case of that for tensor enumerators and hence we omit the proof. We hasten to point out that the components of $A^{(j)}$ and $B^{(j)}$ are polynomials of degree at most $n - 1$, hence their homogenization are of degree exactly $n - 1$. That the overall degree is one less than that of the scalar enumerators is the genesis of an additional factor of $\frac{1}{q}$, which has been incorporated into the map $\Psi$.

**Theorem 21:** Let $E$ be an error basis on $\mathcal{H} = (\mathbb{C}^q)$. Then for any Hermitian operators $M_1, M_2$ on $\mathcal{H} \otimes \mathcal{H}$ and leg $j = 1, \ldots, n$ we have

$$B^{(j)}(w, z; M_1, M_2) = \Psi \left[ A^{(j)} \left( \frac{w+(q^{2-1})z}{q}, \frac{w-z}{q}; M_1, M_2 \right) \right] (IV.30)$$

where $\Psi(e_{E,E'}) = \frac{1}{q} \sum_{E,F} \text{Tr}(F^{1}E^{(w')}F^{(w')}E^{(w')}F^{(w')})e_{E,F'}$. While this theorem states a quantum MacWilliams identity for the vector enumerator analogues of the Shor-Laflamme enumerators, there are vector enumerator extensions of the double and complete enumerators, $C^{(j)}$ and $D^{(j)}$, and $E^{(j)}$ and $F^{(j)}$, are related by MacWilliams identities. The form of these identities, and their proofs, will be covered in §VI below.

The map $\Psi$ is related to a discrete form of the Wigner transform in following sense. Note that the diagonal elements $e_{E,E}$ are left invariant under $\Psi$, as shown by the computation:

$$\Psi(e_{E,E}) = \frac{1}{q} \sum_{E,F'} \text{Tr}(F^{1}E^{(w')}F^{(w')}E^{(w')}F^{(w')})e_{E,F'}.$$

In particular, equation (IV.29) coincides with the discrete Wigner transform as studied in [57] and [58].

Therefore, restricting to the diagonal gives a quantum MacWilliams identity for the reduced enumerators as follows.

**Corollary 22:** Let $E$ be an error basis on $\mathcal{H} = (\mathbb{C}^q)$. Then for any Hermitian operators $M_1, M_2$ on $\mathcal{H} \otimes \mathcal{H}$ and leg $j = 1, \ldots, n$ we have

$$\tilde{B}^{(j)}(w, z; M_1, M_2) = \Psi \left[ \tilde{A}^{(j)} \left( \frac{w+(q^{2-1})z}{q}, \frac{w-z}{q}; M_1, M_2 \right) \right] (IV.30)$$

where $\Psi(e_{E}) = \frac{1}{q} \sum_{F \in E} \omega(E, F)e_{E,F}$ is the discrete Wigner transform on $E$.

In Theorem 14 above we showed $A^{(j)}$ transformed “covariantly” under local unitary transformations. Interestingly, the vector analogue of the quantum MacWilliams identity shows that $B^{(j)}$ transforms “contravariantly” in that one forms the adjoint of the conjugate transformation. This is actually a consequence of the behavior of the map $\Psi$, as shown in the following.

**Lemma 23:** $\Psi \circ \Lambda(U) = \Lambda(U^\dagger) \circ \Psi$ for any unitary $U$.

**Proof:** By direct computation:

$$\Psi[\Lambda(U)(e_{E,E'})] = \sum_{G,G'} \overline{e_{E,G'}} \Psi(e_{G,G'}).$$

$$= \sum_{F,F'} \sum_{G} \text{Tr}(F^{1}G^{(w')}F^{(w')}G^{(w')}F^{(w')})\overline{e_{E,G'}}e_{E,F,F'}.$$

$$= \sum_{F,F'} \text{Tr}[F^{1}(UEU^{\dagger})F^{(w')}U^{(w')}U^{(w')}F^{(w')}F^{(w')}]e_{E,F,F'}.$$

$$= \sum_{F,F'} \text{Tr}[\{U^{(w')}F^{1}U\}E(U^{(w')}F^{1}U)^{\dagger}]e_{E,F,F'}.$$

$$= \sum_{F,F'} \sum_{G,G'} \text{Tr}[G^{1}E^{(w')}F^{(w')}G^{(w')}F^{(w')}G^{(w')}F^{(w')}]e_{E,F,F'}.$$

$$= \sum_{G,G'} \text{Tr}[G^{1}E^{(w')}F^{(w')}G^{(w')}F^{(w')}G^{(w')}F^{(w')}G^{(w')}F^{(w')}]e_{E,F,F'}.$$

$$= \Lambda(U)^{\dagger}(\Psi(e_{G,G'})). \quad (IV.31)$$

**Theorem 24:** Let $U = U_1 \otimes \cdots \otimes U_n$ be a local unitary transformation. Then

$$B^{(j)}(z; U M_1 U^{\dagger}, U M_2 U^{\dagger}) = \Lambda(U^{\dagger}) \left( B^{(j)}(z; M_1, M_2) \right). \quad (IV.32)$$

**Proof:** From the lemma, and Theorems 14 and 21:

$$B^{(j)}(z; U M_1 U^{\dagger}, U M_2 U^{\dagger}) = \Psi \left[ A^{(j)} \left( \frac{w+(q^{2-1})z}{q}, \frac{w-z}{q}; U M_1 U^{\dagger}, U M_2 U^{\dagger} \right) \right]$$

$$= \Psi \left[ \Lambda(U^{\dagger}) A^{(j)} \left( \frac{w+(q^{2-1})z}{q}, \frac{w-z}{q}; M_1, M_2 \right) \right]$$

$$= \Lambda(U^{\dagger}) \Psi \left[ A^{(j)} \left( \frac{w+(q^{2-1})z}{q}, \frac{w-z}{q}; M_1, M_2 \right) \right]$$

$$= \Lambda(U^{\dagger}) B^{(j)}(z; M_1, M_2). \quad (IV.33)$$

V. Tensor Enumerators

In this section we state and prove the main results related to tensor enumerators. Like the vector enumerators of the previous section, these are tensors whose coefficients are drawn from a polynomial ring that depends on the type of enumerator.
under study. We will focus on the tensor enumerator analogues of the Shor-Laflamme enumerators and hence an \( m \)-index tensor enumerator has coefficients in the polynomial ring \( \mathbb{C}[z] \). Our main results parallel those of the previous section, however their proofs are more complicated.

Consider an \( m \)-element subset \( J = \{ j_1, \ldots, j_m \} \subseteq \{ 1, \ldots, n \} \). For \( E \in \mathcal{E}^m \) and \( F \in \mathcal{E}^{n-m} \) let us write \( E \otimes J F \) for the operator where for each \( k = 1, \ldots, m \) we insert \( E_k \) as the \( j_k \)-th factor into \( F \). Recall \( V = \text{span}_{\mathbb{C}} \{ e_E : E, E' \in \mathcal{E} \} \); a degree-\( m \)-tensor enumerator is an element of the \( m \)-fold tensor product of \( V \) (with polynomial coefficients). That is, tensor enumerators lie in \( \mathbb{C}[z] \otimes \mathbb{C} V^{\otimes m} \). We find it convenient to reorder the indexing the basis elements of the tensor product \( V^{\otimes m} \) so that for \( E, E' \in \mathcal{E}^m \) we have
\[
e_{E,E'} = e_E, e_{E_1}' \otimes \cdots \otimes e_{E_m}, e_{E_m}' \in V^{\otimes m}. \tag{V.1}\]

Formally, for a give set of \( m \) legs \( J \), and \( E, E' \in \mathcal{E}^m \), define weight distributions of Hermitian operators \( M_1, M_2 \) on \( \mathcal{S}_r^{\otimes m} \) by
\[
A_d^{(J)}(E, E'; M_1, M_2) = \sum_{F \in \mathcal{E}^{n-m}[d]} \text{Tr}((E \otimes J F)^\dagger M_1) \text{Tr}((E' \otimes J F) M_2), \tag{V.2}\n\]
\[
B_d^{(J)}(E, E'; M_1, M_2) = \sum_{F \in \mathcal{E}^{n-m}[d]} \text{Tr}(E \otimes J F)^\dagger M_1 (E' \otimes J F) M_2. \tag{V.3}\n\]

As with the vector enumerators, weights of \( E \) and \( E' \) are not accounted for in these expressions.

**Definition 25:** Let \( \mathcal{E} \) be an error basis on a Hilbert space \( \mathcal{S}_r \), and \( M_1, M_2 \) Hermitian operators on \( \mathcal{S}_r^{\otimes m} \). We define the tensor enumerators of \( M_1, M_2 \) on set of legs \( J \) by
\[
A^{(J)}(z; M_1, M_2) = \sum_{E, E' \in \mathcal{E}^m} \sum_{d=0}^{n-m} A_d^{(J)}(E, E'; M_1, M_2) z^d e_{E, E'}, \tag{V.4}\n\]
\[
B^{(J)}(z; M_1, M_2) = \sum_{E, E' \in \mathcal{E}^m} \sum_{d=0}^{n-m} B_d^{(J)}(E, E'; M_1, M_2) z^d e_{E, E'}. \tag{V.5}\n\]

where \( A_d^{(J)} \) and \( B_d^{(J)} \) are defined in (V.2) and (V.3).

We can extend the weighted trace in Definition 11 above to link tensor enumerators of different degrees. Let \( K \subseteq J \), with \( |K| = k \) and \( |J| = m \). Then any element of \( \mathcal{E}^m \) can be written as \( F \otimes K E \) for \( F \in \mathcal{E}^k \) and \( E \in \mathcal{E}^{m-k} \). We define \( \text{Tr}^{J}_K : \mathbb{C}[z] \otimes \mathbb{C} V^{\otimes m} \to \mathbb{C}[z] \otimes \mathbb{C} V^{\otimes k} \) by linearly extending
\[
\text{Tr}^{J}_K(\epsilon_{F \otimes K E, F' \otimes K E'}) = \begin{cases} z^{\text{wt}(E)} e_{E, F}, & \text{if } E = E' \\ 0, & \text{if } E \neq E' \end{cases} \tag{V.6}\n\]
to polynomial coefficients. The proof of the following result is straightforward computation and so is left for the reader.

**Proposition 26:** Let \( \mathcal{E} \) be an error basis on a Hilbert space \( \mathcal{S}_r \), and \( M_1, M_2 \) Hermitian operators on \( \mathcal{S}_r^{\otimes n} \). Then for any \( K \subseteq J \). We have \( \text{Tr}^{J}_K(A^{(J)}(z; M_1, M_2)) = A^{(K)}(z; M_1, M_2) \), and similarly \( \text{Tr}^{J}_K(B^{(J)}(z; M_1, M_2)) = B^{(K)}(z; M_1, M_2) \).

In Definition 13 above, we used the natural unitary action on the error basis to define a representation \( \Lambda : U(\mathcal{S}_r) \to \mathcal{L}(V) \). This extends canonically to unitary operators of \( \mathcal{S}_r^{\otimes m} \) (not just local unitary operators). Namely, just as in (IV.7) we can define \( c_{EF} \) by \( U^\dagger EU = \sum_{G,G' \in \mathcal{E}^m} c_{EG} c_{EG'} G G' \), where now \( U \in U(\mathcal{S}_r^{\otimes m}) \) and \( E \in \mathcal{E}^m \), and form (without additional decoration) \( \Lambda(U(\mathcal{S}_r^{\otimes m})) \to \mathcal{L}(V^{\otimes m}) \) by
\[
\Lambda(U)(e_{E,E'}) = \sum_{G,G' \in \mathcal{E}^m} c_{EG} c_{EG'} e_{G,G'}. \tag{V.7}\n\]

Now, we have \( c_{EG} = \frac{1}{q} \text{Tr}(G^\dagger U EU) \), and in the previous section we find \( \overline{\text{Tr}}_{EF} = \frac{1}{q} \text{Tr}(E^\dagger UEU) \) and thus
\[
\Lambda(U^\dagger)(e_{E,E'}) = \sum_{G,G' \in \mathcal{E}} e_{G,G'} c_{EG} c_{EG'} \overline{e}_{E,E'}. \tag{V.8}\n\]

**Theorem 27:** Let \( \mathcal{E} \) be an error basis on a Hilbert space \( \mathcal{S}_r \), \( U = U_1 \otimes \cdots \otimes U_n \) be a local unitary operator, and \( M_1, M_2 \) be Hermitian operators on \( \mathcal{S}_r^{\otimes n} \). Then for any subset of legs \( J \subseteq \{ 1, \ldots, n \} \) we have
\[
A^{(J)}(z; U M_1 U^\dagger, U M_2 U^\dagger) = \Lambda(U) \left( A^{(J)}(z; M_1, M_2) \right), \tag{V.9}\n\]
where \( U_J = \bigotimes_{j \in J} U_j \).

**Proof:** We write
\[
A^{(J)}(z; U M_1 U^\dagger, U M_2 U^\dagger) = \sum_{E,E'} \sum_{F} \text{Tr}((E \otimes J F)^\dagger U M_1 U^\dagger) . \tag{V.10}\n\]
\[
\cdot \text{Tr}((E' \otimes J F) U M_2 U^\dagger) z^{\text{wt}(F)} e_{E,E'} . \tag{V.10}\n\]
\[
\sum_{E,E'} \sum_{F} \text{Tr}((U_j^\dagger E U_j \otimes J \hat{U} F U^\dagger) M_1) . \tag{V.10}\n\]
\[
\cdot \text{Tr}((U_j^\dagger E U_j \otimes J \hat{U} F U^\dagger) M_2) z^{\text{wt}(F)} e_{E,E'} . \tag{V.10}\n\]

where \( U_J = \bigotimes_{j \in J} U_j \) and \( \hat{U} = \bigotimes_{j \in J^c} U_j \). The invariance of this expression under \( \hat{U} \) follows exactly as in Theorem 4. For \( U_J \) we expand as above
\[
A^{(J)}(z; U M_1 U^\dagger, U M_2 U^\dagger) = \sum_{E,E',G,G'} \sum_{F} \overline{\text{Tr}}_{EG} \overline{e}_{G,G'} \text{Tr}((G \otimes J F)^\dagger M_1) . \tag{V.11}\n\]
\[
\cdot \text{Tr}((G' \otimes J F) M_2) z^{\text{wt}(F)} e_{E,E'} . \tag{V.11}\n\]
\[
= \sum_{G,G'} \sum_{F} \text{Tr}((G \otimes J F)^\dagger M_1) \text{Tr}((G' \otimes J F) M_2) z^{\text{wt}(F)} e_{E,E'} . \tag{V.11}\n\]
\[
= \Lambda(U) \left( A^{(J)}(z; M_1, M_2) \right). \tag{V.11}\n\]

We can also prove the weighted trace (V.6) intertwines the representation \( \Lambda \) on different degree tensors as follows.

**Proposition 28:** Let \( U = U_1 \otimes \cdots \otimes U_n \) be a local unitary operator on \( \mathcal{S}_r^{\otimes n} \), and let \( K \subseteq J \subseteq \{ 1, \ldots, n \} \). Then
\[
\Lambda(U_K) \cdot \text{Tr}^{J}_K = \text{Tr}^{J}_K \circ \Lambda(U_J). \tag{V.11}\n\]
Therefore
\[
\Lambda(U_K)[\text{tr}_K(e_{E'\otimes K,F,F'\otimes K,F'})] = z^{wt(F)}\Lambda(U_K)(e_{E,F})
\]
while \(\Lambda(U_K)[\text{tr}_K(e_{E'\otimes K,F,F'\otimes K,F'})] = 0\) when \(F \neq F'\).

Now we compute
\[
\text{tr}_K[\Lambda(U_J)(e_{E'\otimes K,F,F'\otimes K,F'})] = \sum_{G,G',H,H'} \overline{e}_{EG}c_{EG'}e_{G,G'},
\]
\[
\text{tr}(E_{H,F})^\dagger \text{tr}(E_{H,F})^\dagger = \sum_{H \in E^{m-k}} z^{wt(H)}c_{EH}c_{F'H} = \sum_{G,G' \in E^k} \overline{e}_{EG}c_{EG'}e_{G,G'},
\]
\[
(V.12)
\]

From Lemma 3,
\[
\sum_{H \in E^{m-k}} \text{tr}(U_1^\dagger U_1^\dagger U_1^\dagger U_1^\dagger) = q^{2(m-k)} \sum_{H \in E^{m-k}[d]} c_{EH}c_{F'H}
\]
\[
= \left\{ \begin{array}{ll}
q^{2(m-k)} & \text{if } F = F' \in E^{m-k}[d] \\
0 & \text{otherwise.}
\end{array} \right. 
(V.14)
\]

Therefore
\[
\sum_{H \in E^{m-k}} z^{wt(H)}c_{EH}c_{F'H} = \left\{ \begin{array}{ll}
z^{wt(F)} & \text{if } F = F' \\
0 & \text{otherwise,}
\end{array} \right. 
(V.15)
\]
and so the right side of the equation coincides with the left on every element of \(V^{\otimes m}\).

A crucial property we will need is that tensor enumerators are homomorphic under the tensor product. That is, the tensor enumerator of a tensor product is the tensor product of the individual enumerators. Formally this captured in the following result. Note that when one of \(J\) or \(K\) is empty, the corresponding tensor enumerator is a scalar enumerator and so the tensor product on the right of (V.16) is the ordinary product. In this way the following results covers Proposition 15 as a special case.

Proposition 29: Let \(M_1, M_2\) and \(N_1, N_2\) be Hermitian operators on Hilbert spaces \(F_i\) and \(F_i\) respectively, and let \(J\) and \(K\) be subsets of legs on each of these spaces. Then
\[
\Lambda(J\otimes K)(z; M_1 \otimes N_1); M_2 \otimes N_2) = \Lambda(J)(z; M_1, M_2) \otimes \Lambda(K)(z; N_1, N_2),
\]
\[
(V.16)
\]
and similarly for \(\Lambda\).

Proof: Just as Lemma 5 this is a direct computation:
\[
\Lambda(J\otimes K)(z; M_1 \otimes N_1); M_2 \otimes N_2) = \sum_{E, E', F, F'} \sum_{G, H} \text{Tr} \left[ (E \otimes J G) \otimes (F \otimes K H) \right] (M_1 \otimes N_1) \cdot \text{Tr} \left[ (E' \otimes K H) \right] (M_2 \otimes N_2) \cdot (E \otimes F)^{wt(J)} (F' \otimes F)^{wt(K)}
\]
\[
= \sum_{E, E', F, F'} \text{Tr} \left[ (E \otimes J G) \otimes (F \otimes K H) \right] \text{Tr} \left[ (E' \otimes K H) \right] (M_1 \otimes N_1) \cdot \text{Tr} \left[ (F' \otimes F' \otimes H) \right] (M_2 \otimes N_2)
\]
\[
= \sum_{E, E', F, F'} \text{Tr} \left[ (E \otimes J G) \otimes (F \otimes K H) \right] \cdot \text{Tr} \left[ (F' \otimes F' \otimes H) \right] \cdot (E \otimes F)^{wt(J)} (F' \otimes F')
\]
\[
(V.17)
\]

In Lemma 17 above, we showed that a nonzero off-diagonal term in a vector enumerator indicates the existence of a stabilizer supported on the leg of the vector enumerator. This results carries over to general tensor enumerators with an identical proof, which we state below for completeness. However unlike in Theorem 18 for vector enumerators, we do not expect tensor enumerators with off-diagonal terms to factor; the proof of that theorem relied critically on the fact that any code stabilized by a Pauli of the form \(P \otimes J \otimes F^{(n-1)}\) must have a separable factor.

Proposition 30: Let \(\mathcal{C}\) be a \([n, k]\) stabilizer code and \(J\) be a set of legs of size \(m\). Then for \(P \neq P' \in P^m\),
1) \(\Lambda(J)(P, P')\) is nonzero if and only if \(P' \otimes J \otimes F^{(n-m)}\) is a stabilizer;
2) \(\Lambda(J)(P, P')\) is nonzero if and only if \(\Lambda(J)(P, P')\) is nonzero.

Exactly as with vector enumerators, the logical legs of the encoding map of a quantum code can be cleaned and therefore its tensor enumerator must be diagonal. This follows from the above proposition. In fact, the components of the enumerator contain the Shor-Laflamme enumerators of the code.

Proposition 31: Let \(\mathcal{C}\) be a \([n, k]\) binary quantum code and \(|\mathcal{C}\rangle\) the state associated to its encoding map. Let \(B\) be the legs of \(|\mathcal{C}\rangle\) corresponding to the logical operators, and \(\Pi_\mathcal{C}\) the orthogonal projection onto \(\mathcal{C}\). Then
\[
A(\Pi_\mathcal{C}) = 4^k \cdot \Lambda(J)(I, I; |\mathcal{C}\rangle\langle \mathcal{C}|),
\]
\[
B(\Pi_\mathcal{C}) = 2^k \cdot \sum_{P \in P^k} \Lambda(J)(P, P; |\mathcal{C}\rangle\langle \mathcal{C}|). 
(V.18)
\]
Finally, we state the MacWilliams identity for tensor enumerators. This includes as a special case the MacWilliams identities for scalar and vector enumerators, and is in turn a special case of a general framework for quantum MacWilliams identities covered in the next section. Specifically the following theorem is a special case of Corollary 41 below.

Theorem 32: Let \(\mathcal{E}\) be an error basis on \(F_i = C^q\), and \(M_1, M_2\) Hermitian operators on \(F_i^{\otimes n}\). Let \(J \subseteq \{1, \ldots, n\}\) be a subset of size \(m\). Then
\[
\Lambda(J)(w; M_1, M_2) = \Psi\left[ \Lambda(J) \left( \frac{w+q^2-1}{q}; \frac{w+q^2}{q}; M_1, M_2 \right) \right]
\]
\[
(V.20)
\]
where \(\Psi(e_{E,F,F'}) = \frac{1}{q^{2m}} \sum_{E, F, F' \in E^m} \text{Tr} \left[ (E' \otimes F' \otimes F') \right] (E \otimes F)^{wt(J)} \cdot (E' \otimes F)^{wt(J)}
\]

Just as we did with vector enumerators, we can use the MacWilliams identity to prove the contravariant behavior of the \(B\)-tensor enumerator.

Corollary 33: Let \(U = U_1 \otimes \cdots \otimes U_n\) be a local unitary operator on \(F_i^{\otimes n}\), and let \(J \subseteq \{1, \ldots, n\}\) be any subset of
legs. Then
\[
B^{(J)}(z; UM_1 U^\dagger, UM_2 U^\dagger) = \Lambda(U_j^*) \left( B^{(J)}(z; M_1, M_2) \right),
\]
(V.21)
where \( U_j = \bigotimes_{j \in J} U_j \).

**Proof:** From the theorem, Theorem 27, and Lemma 23 we have
\[
B^{(J)}(w, z; UM_1 U^\dagger, UM_2 U^\dagger)
= \Psi \left[ \Lambda(U_j) A^{(J)} \left( \frac{w + (q^2 - 1)z}{q}, \frac{w-z}{q}; UM_1 U^\dagger, UM_2 U^\dagger \right) \right]
= \Lambda(U_j) \Psi \left[ A^{(J)} \left( \frac{w + (q^2 - 1)z}{q}, \frac{w-z}{q}; M_1, M_2 \right) \right]
= \Lambda(U_j) \left( B^{(J)}(z; M_1, M_2) \right).
\]
(V.22)

VI. ALL QUANTUM MACWILLIAMS IDENTITIES

In this section we provide a unified proof of all the quantum MacWilliams identities, including those provided in [35], [37], [39], [40], and [43], as well as that of the vector and tensor enumerators (Theorem 21 and Theorem 32) above. The key idea is to identify the critical relationship that the error basis and the weight function must satisfy, (VI.7), in order to link the \( A \)-type and \( B \)-type enumerators. In application, one then simply verifies this relationship holds for the error basis and weight function at hand.

Let \( \mathcal{E} \) be an error basis on \( \mathfrak{H} \). In this section we define a general notion of weight function on \( \mathcal{E} \) that includes the usual Hamming weight, double and complete weights, and vector and tensor weights from above as special cases. To capture vector and tensor weights, our weight functions will need to be on two variables \( \langle E, E' \rangle \in \mathbb{E}^2 \); but, to simultaneously treat the Hamming, double, or complete weights we need to force a weight function to be restricted to the diagonal \( \langle E, E \rangle \).

In order to accomplish this we borrow the logical symbol “\( \perp \)” to represent undefined. We then provide a unified proof of all the MacWilliams identities referred to previously, and recover each as a corollary to this theorem.

**Definition 34:** A weight function is defined as any function \( \text{wt} : \mathcal{E}^2 \to \mathbb{Z}_{\geq 0} \cup \{ \perp \} \). A weight function is scalar if it is supported on the diagonal: \( \text{wt}(E, E') = \perp \) whenever \( E \neq E' \).

Given a tuple \( (\text{wt}_1, \ldots, \text{wt}_n) \) of such functions, we define \( \text{wt} : \mathcal{E}^{2n} \to \mathbb{Z}_{\geq 0} \cup \{ \perp \} \) by
\[
\text{wt}(E_1 \otimes E'_1 \otimes \cdots \otimes E_n \otimes E'_n) = \text{wt}_1(E_1, E'_1) + \cdots + \text{wt}_n(E_n, E'_n),
\]
(VI.1)
where we define \( t + \perp = \perp \) for any \( t \in \mathbb{Z}_{\geq 0} \cup \{ \perp \} \).

For a \( k \)-tuple of indeterminates \( \mathbf{u} = (u_1, \ldots, u_k) \) we write
\[
\mathbf{u}^{\text{wt}(E,E')} = \begin{cases} 
\prod_{j=1}^k u_j^{\text{wt}(E_j, E'_j)} & \text{if } \text{wt}(E, E') \neq \perp \\
0 & \text{if } \text{wt}(E, E') = \perp.
\end{cases}
\]
(VI.2)

Here \( \text{wt}(E, E')_j \) is the \( j \)-th coordinate of \( \text{wt}(E, E') \).

For example, the usual (homogeneous) quantum weight enumerator uses variables \( (w, z) \) and has weight function
\[
\text{wt}(E, E') = \begin{cases} 
(1, 0) & \text{if } E = E' = I \\
(0, 1) & \text{if } E = E' \neq I \\
\perp & \text{if } E \neq E'.
\end{cases}
\]
(VI.3)

The (homogeneous) double enumerator for the usual Pauli basis uses variables \( (w, x, y, z) \) and has
\[
\text{wt}(E, E') = \begin{cases} 
(1, 0, 1, 0) & \text{if } E = E' = I \\
(0, 1, 1, 0) & \text{if } E = E' = X \\
(1, 0, 0, 1) & \text{if } E = E' = Y \\
(1, 0, 0, 1) & \text{if } E = E' = Z \\
\perp & \text{if } E \neq E'.
\end{cases}
\]
(VI.4)

The complete enumerator of an error basis \( \mathcal{E} \) has variables \( \{ u_E \}_{E \in \mathcal{E}} \) and the weight \( \text{wt}(E, E) \) has a 1 in the \( E \)-th position and zero elsewhere, while for \( E \neq E' \) the weight \( \text{wt}(E, E') = \perp \).

For vector enumerators, their weight functions are composed from two weight functions. In all but one coordinate we have a scalar weight function, such as any of those above. In the one remaining coordinate we have variables \( \{ e_{E,E'} \}_{E,E' \in \mathcal{E}} \) and the weight function where \( \text{wt}(E, E') \) equal to 1 in the \( (E, E') \) position and zero elsewhere. Hence, the overall set of variables consists of the scalar variables and \( \{ e_{E,E'} \} \). The latter set of variables has the feature that every monomial has precisely one of these appearing linearly, hence our interpretation as a vector with polynomial coefficients.

Tensor enumerators are natural extensions of vector enumerators, where in each leg of the tensor we take a distinct copy of the weight function \( \text{wt}(E, E') \) above. That is for an order \( m \) tensor, we have variables \( \{ e_{E_1,E_1'}, \ldots, e_{E_m,E_m'} \} \) and the weight function in the \( j \)-th leg of the tensor is \( \text{wt}(E_j, E'_j) \) that has a 1 is \( (E_j, E'_j) \) position for variable \( e_{E_j,E'_j} \) and zero elsewhere.

We define enumerators of Hermitian operators \( M_1, M_2 \) over a weight function \( \text{wt} \) as
\[
A(u; M_1, M_2) = \sum_{E,E' \in \mathcal{E}^n} \text{Tr}(E^\dagger M_1 E M'_2) u^{\text{wt}(E,E')} \tag{VI.5}
\]
\[
B(u; M_1, M_2) = \sum_{E,E' \in \mathcal{E}^n} \text{Tr}(E^\dagger M_1 E'_2 M'_2) u^{\text{wt}(E,E')} \tag{VI.6}
\]

This recovers scalar, vector, and tensor enumerators over any of weight functions given above as special cases.

**Theorem 35:** Let \( \mathcal{E} \) be an error basis on \( \mathfrak{H} \) and let \( \text{wt} \) be associated to weight functions \( \{ \text{wt}_1, \ldots, \text{wt}_n \} \), where each \( \text{wt}_j : \mathcal{E}^2 \to \mathbb{Z}_{\geq 0} \cup \{ \perp \} \). Suppose there exists an algebraic mapping \( \Phi(\mathbf{u}) = (\Phi_1(\mathbf{u}), \ldots, \Phi_n(\mathbf{u})) \) where each \( j = 1, \ldots, n \) has:
\[
\Phi(\mathbf{u})^{\text{wt}_j(D,D')} = \frac{1}{q^2} \sum_{E,E'} \text{Tr}(E^\dagger D'E'(D')^\dagger) u^{\text{wt}_j(E,E')} \tag{VI.7}
\]
Then the enumerators over this weight function satisfy
\[ B(u; M_1, M_2) = A(\Phi(u); M_1, M_2). \]  

Proof: We begin by expressing
\[ M_1 = \frac{1}{q^n} \sum_{D \in \mathcal{E}^n} \text{Tr}(D^1 M_1) D, \]  
\[ M_2 = \frac{1}{q^n} \sum_{D' \in \mathcal{E}^n} \text{Tr}(D'M_2)(D')^\dagger. \]

Then
\[ B(u; M_1, M_2) = \sum_{E, E' \in \mathcal{E}^n} \text{Tr}(E^1 M_1 E'M_2) u^{\text{wt}(E, E')} = \frac{1}{q^n} \sum_{D, D', E, E' \in \mathcal{E}^n} (\text{Tr}(D^1 M_1) \text{Tr}(D'M_2) \cdot \text{Tr}(E^1 D'E'(D')^\dagger)) u^{\text{wt}(E)}. \]  

Therefore,
\[ B(u; M_1, M_2) = \sum_{D, D' \in \mathcal{E}^n} \text{Tr}(D^1 M_1) \text{Tr}(D'M_2) \Phi(u)^{\text{wt}(D, D')} = A(\Phi(u); M_1, M_2). \]

Proof: We can reduce from the complete enumerator of the previous result by setting \( u_I = w \) and \( u_D = z \) for \( D \neq I \). So in the case \( D = D' = I \) we have
\[ \Phi_I(w, z) = \frac{1}{q} \sum_{E \in \mathcal{E}} u_E = \frac{w + (q^2 - 1)z}{q}. \]

While for any \( D = D' \neq I \) we have
\[ \Phi_D(w, z) = \frac{1}{q} \sum_{E} \omega(D, E)u_E = \frac{w - z}{q}. \]

All \( \Phi_D(w, z) \) are equal, so \( \Phi(w, z) = (\Phi_I(w, z), \Phi_D(w, z)) \) is well defined and satisfies condition (VI.7). □

The double weight enumerator, for both the binary [39] and non-binary [40] Pauli bases, are defined over homogeneous variables \( (w, x, y, z) \). When \( q > 2 \), we can define a refined double weight enumerator over variables \( u = (x, z) = (x_0, \cdots, x_{q-1}, z_0, \cdots, z_{q-1}) \) using the scalar weight function implicitly defined via
\[ (x, z)^{\text{wt}(X^a z^b, X^c z^d)} = x_a z_b \]  
and zero otherwise. This gives us a new MacWilliams identity related to that of the double enumerators as follows.

Lemma 38: For the Pauli basis \( \mathcal{P} \) on \( \mathcal{H} = \mathbb{C}^q \), let \( (x, z) \) be variables and \( \text{wt} \) the weight function as defined above. Then their enumerators \( A, B \) satisfy
\[ B(u; z_b; M_1, M_2) = A(\frac{1}{\sqrt{q}} \sum_{d} e^{\pi i d q} x_d, \frac{1}{\sqrt{q}} \sum_{c} e^{-\pi i c q} z_c; M_1, M_2), \]  

where \( \zeta = e^{2\pi i / q} \).

Proof: Write \( \Phi(x, z) = (\Theta_a(x, z), \Psi_b(x, z)) \). When \( D = D' = X^a Z^b \), using (VI.20), we require
\[ (\Phi(x, z)^{\text{wt}(X^a z^b, X^c z^d)} = \Theta_a(x, z) \cdot \Psi_b(x, z) \]
We applying this reduction we find of these provide a well defined transform that satisfies (VI.7). Θ

identity, hence for a, b > 0, any choice remains the same for any a, b > 0, any choice of these provide a well defined transform that satisfies (VI.7).

The following is just a specialization of Corollary 36 to the Pauli error basis.

Corollary 40 [39]: For the Pauli basis P on $\mathcal{H} = \mathbb{C}^q$, the complete quantum weight enumerators satisfy

\[ F(u_{ab}; M_1, M_2) = E(\frac{1}{q} \sum_{c,d} \zeta^{ad-bc} u_{cd}; M_1, M_2) \]

where $\zeta = e^{2\pi i/q}$.

Proof: The complete weight function is essentially the identity, hence for $D = D' = X^a Z^b$ we require we require

$$\Phi(u)^{wt(X^a Z^b, X^a Z^b)} = \Phi_{(a,b)}(u)$$

When $D \neq D'$ we need $0 = 0$, which always holds. □

Finally we consider the vector and tensor enumerators over any scalar system that satisfies a MacWilliams identity, such as any of those above.

Corollary 41: Let $E$ be an error basis on $\mathcal{H} = \mathbb{C}^q$ and $\tilde{w}$ be any scalar weight function that satisfies condition (VI.7) of Theorem 35 for algebraic map $\Phi(u)$. Let $J = \{j_1, \ldots, j_m\} \subseteq \{1, \ldots, n\}$ and define $wt$ on $(E \times E)^n$ implicitly by

$$\Psi(E, E') = \bigg( \prod_{k \notin J} u_{\tilde{v}_k}(E_k, E'_k) \bigg)_{E_1, E'_1, \ldots, E_m, E'_m} \quad (VI.34)$$

where each set of variables $e_j = \{e_{E,E'}\} \forall E,E' \in \mathcal{E}$. Then the enumerators of this weight function are the tensor enumerators, $A = A^{(J)}$ and $B = B^{(J)}$, and

$$B^{(J)}(u; M_1, M_2) = \Psi[A^{(J)}(\Phi(u); M_1, M_2)] \quad (VI.35)$$

where $\Psi(e_{E,E'}) = \frac{1}{q^m} \sum_{F \in \mathcal{E}} \text{Tr}(F^\dagger EF'(E')^\dagger) e_{F,F'}$.

Proof: For the legs not in $J$, condition (VI.7) of the theorem is satisfied by assumption. For the legs of $J$, we have $\Psi(e_{E,E'}) = \frac{1}{q^m} \sum_{F \in \mathcal{E}} \text{Tr}(F^\dagger EF'(E')^\dagger) e_{F,F'}$, which is precisely condition (VI.7) of the theorem. □

VII. Tensor Networks

Here we show the main application of this work: when a quantum code is created via the lego formalism of [31], we can compute its enumerators by tracing the tensor enumerators of the constituent building blocks via precisely the same tensor network.

Recall that a tensor network state is expressed in the computational basis of $(\mathbb{C}^q)^{\otimes n}$ as

$$|T\rangle = \sum_{j_1, \ldots, j_n} c_{j_1, \ldots, j_n} |j_1, \ldots, j_n\rangle \quad (VII.1)$$

and we define its trace on, say, legs 1 and 2 to be

$$\wedge_{1,2} |T\rangle \propto \sum_j \sum_{j_3, \ldots, j_n} c_{j, j_3, \ldots, j_n} |j_3, \ldots, j_n\rangle \quad (VII.2)$$

Note that we may need to renormalize to get a state of norm one. Writing $|\beta\rangle = \frac{1}{\sqrt{q}} \sum_j |jj\rangle$, we can informally write

$$\wedge_{1,2} |T\rangle = (|\beta\rangle \otimes I^{\otimes n-2}) |T\rangle \quad (VII.3)$$

or more generally $\wedge_{j,k} |T\rangle = (|\beta\rangle \otimes |j,k\rangle I^{\otimes n-2}) |T\rangle$ using the notation of the previous sections.

We extend the above trace on legs $1 \leq j < k \leq n$ to a general Hermitian operator $M$ on $(\mathbb{C}^q)^{\otimes n}$ as

$$\wedge_{j,k} M = (|\beta\rangle \otimes |j,k\rangle I^{\otimes n-2}) M (|\beta\rangle \otimes |j,k\rangle I^{\otimes n-2}) \quad (VII.4)$$

which is then a Hermitian operator on $(\mathbb{C}^q)^{\otimes (n-2)}$. We then have

$$\text{Tr}(A (\wedge_{j,k} M)) = \text{Tr}((|\beta\rangle \otimes |j,k\rangle A) M) \quad (VII.5)$$
for any operator $A$ on $(C^q)^{\otimes (n-2)}$, which shows the trace on legs $j, k$ can be viewed as the dual operation to tensor product with respect to a Bell state.

Now consider any error basis $E$ on $\mathcal{H} = \mathbb{C}^q$. Write $E^*$ for the element-wise complex conjugate of $E$. Clearly as $EF = \omega(E,F)FE$ we have $E^*F^* = \omega(E,F)^{-1}F^*E^*$, and thus for every $E, F$ we have that $E \otimes E^*$ commutes with $F \otimes F^*$.

Lemma 42: For any error basis $E$ on $\mathcal{H} = \mathbb{C}^q$ we have

$$|\beta\rangle\langle\beta| = \frac{1}{q^2} \sum_{E \in \mathcal{E}} E \otimes E^*. \quad (\text{VII.6})$$

Proof: As $|\beta\rangle\langle\beta|$ is an operator on $\mathcal{H} \otimes \mathcal{H}$ we can write

$$|\beta\rangle\langle\beta| = \sum_{E,F} c_{E,F} E \otimes F. \quad (\text{VII.7})$$

But,

$$c_{E,F} = \frac{1}{q^2} \text{Tr}(|\beta\rangle\langle\beta| (E \otimes F)^\dagger)$$

$$= \frac{1}{q^2} \sum_{j,k} \langle j, j| E^* \otimes F^* | k, k \rangle$$

$$= \frac{1}{q^2} \sum_j (k| E^* j \rangle \langle j | F^* | k \rangle)$$

$$= \frac{1}{q^2} \text{Tr}(E^* F^*) = \left\{ \begin{array}{ll} \frac{1}{q^2} & \text{if } F = E^* \\ 0 & \text{otherwise.} \end{array} \right. \quad (\text{VII.8})$$

Recall that tensor enumerators lie in

$$\mathbb{C}[z] \otimes \mathbb{C} V^{\otimes m} = \text{span}_{\mathbb{C}[z]} \{ e_{E,F} : E, E' \in \mathbb{C}^{\otimes m} \}. \quad (\text{VII.9})$$

We define the trace on legs $j, k$ on this space as

$$\wedge^{j,k} e_{E,E'} = e_{E \setminus \{ E_j, E_k \}, E' \setminus \{ E'_j, E'_k \}} \quad (\text{VII.10})$$

if $E_k = E'_k$ and $E_j = (E'_j)^*$, and zero otherwise. We extend linearly to all of $\mathbb{C}[z] \otimes \mathbb{C} V^{\otimes m}$. Theorem 43: Suppose $j, k \in J \subseteq \{1, \ldots, m\}$. Then

$$\wedge^{j,k} A^{(J)}(z; M_1, M_2) = A^{(J \setminus \{ j,k \})}(z; \wedge^{j,k} M_1, \wedge^{j,k} M_2), \quad (\text{VII.11})$$

and similarly for $B^{(J)}$.

Proof: For $A^{(J)}$ we directly compute,

$$\wedge^{j,k} A^{(J)}(z; M_1, M_2)$$

$$= \sum_{E, E', F} \left[ \text{Tr}((E \otimes J F)^\dagger M_1) \text{Tr}((E' \otimes J F) M_2) \right]$$

$$\cdot z^{\wedge^{j,k} e_{E,E'}} \quad (\text{VII.12})$$

$$= \sum_{E} \sum_{E'} \sum_{G, G'} \left\{ \text{Tr}(((G \otimes G^*) \otimes J k \tilde{E} \otimes J F)^\dagger M_1) \right\}$$

$$\cdot \text{Tr}(((G' \otimes (G^*)^*) \otimes J k \tilde{E}' \otimes J F) M_2) \right\}$$

$$\cdot z^{\wedge^{j,k} e_{E,E'}} \quad (\text{VII.13})$$

$$= \sum_{E, E', F} \left[ \text{Tr}((\tilde{E} \otimes J \{ j,k \} F)^\dagger (\wedge^{j,k} M_1)) \right]$$

$$\cdot \text{Tr}((\tilde{E}' \otimes J \{ j,k \} F) (\wedge^{j,k} M_2))$$

$$\cdot z^{\wedge^{j,k} e_{E,E'}} \quad (\text{VII.14})$$

where we write $\tilde{E} = E \setminus \{ E_j, E_k \}$ and similarly for $\tilde{E}'$. For $B^{(J)}$ we appeal to the MacWilliams identity. Recall the generalized discrete Wigner transform:

$$\Psi^{(m)}(e_{E,E'}) = \frac{1}{q^{2m}} \sum_{F,F'} \text{Tr}(F^\dagger EF'(E')^\dagger) e_{F,F'} \quad (\text{VII.16})$$

Here we decorate the transform as we will be working in multiple degrees. We have

$$\wedge^{j,k} \Psi^{(m)}(e_{E,E'}) = \frac{1}{q^{2m}} \sum_{F,F'} \text{Tr}(F^\dagger EF'(E')^\dagger)(\wedge^{j,k} e_{F,F'})$$

$$= \frac{1}{q^{2m}} \sum_{F,F'} \text{Tr}(F^\dagger E_j F'(E'_j)^\dagger) \text{Tr}(F^{j*} E_k (F')^* (E'_k)^\dagger)$$

$$\cdot \sum_{F,F'} \text{Tr}(\tilde{E} \tilde{E}' (\tilde{E}')^\dagger) e_{F,F'}$$

$$= \frac{1}{q^{m}} \sum_{F,F'} \text{Tr}(F^\dagger E_j F'(E'_j)^\dagger) \text{Tr}(F^{j*} E_k (F')^* (E'_k)^\dagger)$$

$$\cdot \Psi^{(m-2)}(e_{E,E'}) \quad (\text{VII.17})$$

where like above we have written $\tilde{E} = E \setminus \{ E_j, E_k \}$, and similarly for $F, F'$, $E', F'$. Now,

$$\sum_{F,F'} \text{Tr}(F^\dagger E_j F'(E'_j)^\dagger) \text{Tr}(F^{j*} E_k (F')^* (E'_k)^\dagger)$$

$$= \sum_{F,F'} \text{Tr}((F \otimes F^*)^\dagger (E_j \otimes E_k) (F' \otimes (F')^*) (E'_j \otimes E'_k)^\dagger)$$

$$= q^4 \text{Tr}(|\beta\rangle\langle\beta| (E_j \otimes E_k)|\beta\rangle\langle\beta| (E'_j \otimes E'_k))^\dagger$$

$$= q^4 \langle \beta| E_j \otimes E_k |\beta\rangle \langle \beta| E'_j \otimes E'_k |\beta\rangle. \quad (\text{VII.18})$$

But just as we have above

$$\langle \beta| E_j \otimes E_k |\beta\rangle = \frac{1}{q} \sum_{r,s} \langle r, r| E_j \otimes E_k |s, s \rangle$$

$$= \frac{1}{q} \sum_{r,s} \langle s| E_j^r |r \rangle \langle r | E_k |s \rangle$$

$$= \frac{1}{q} \text{Tr}(E_j^r E_k) \quad (\text{VII.19})$$

Thus, $e_{E,E'} = \wedge^{j,k} e_{E,E'}$ and

$$\wedge^{j,k} [\Psi^{(m)}(e_{E,E'})] = \Psi^{(m-2)}(e_{E,E'}) = \Psi^{(m-2)}(\wedge^{j,k} e_{E,E'}) \quad (\text{VII.20})$$

Therefore,

$$\wedge^{j,k} B^{(J)}(w; z; M_1, M_2)$$

$$= \wedge^{j,k} \Psi^{(m)} \left[ A^{(J)}(w + \frac{q^2 - 1}{q} z, w - z; M_1, M_2) \right]$$

$$= \Psi^{(m-2)} \left[ A^{(J \setminus \{ j,k \})}(w + \frac{q^2 - 1}{q} z, w - z; \wedge^{j,k} M_1, \wedge^{j,k} M_2) \right]$$

$$= B^{(J \setminus \{ j,k \})}(w, z; \wedge^{j,k} M_1, \wedge^{j,k} M_2). \quad (\text{VII.21})$$
Example 44: Consider the holographic code from [21] as shown in Figure 1 above. This five logical qubit shallow holographic code only has 20 physical qubits, and so it would not be numerically challenging to compute its enumerators directly from the definitions. However, it is an informative example that we can illustrate much more easily using tensor tracing two tensor enumerators with three legs.

Namely, there are only three legs across the blue cut in the figure and hence we can compute the three index tensor enumerator above the cut separately. The left Bacon-Shor code has tensor enumerator

$$A^{[[4,1,2]]}_{[[3,4]]} = e_{1,1} + z^2 e_{1,3} + z^2 e_{3,1} + z^2 e_{3,3} + z^2 e_{5,1} + z^2 e_{5,3} + z e_{2,4} + z e_{4,2} + z e_{4,4} + z e_{6,2} + z e_{6,6} + z^2 e_{2,2} + z^2 e_{4,4} + z^2 e_{6,6}.$$  

(VII.22)

Note that the two legs on the boundary will never be traced, and so we need only track the tensor for remaining two legs, legs 3 and 4 by our counting. The tensor enumerator for the perfect tensor with legs 4, 5, and 6 has 64 nonzero terms and so we will not write out $A^{[[4,5,6]]}$ explicitly, however each coefficient is simply a single monomial. Tracing leg 3 of $A^{[[3,4]]}_{[[4,1,2]]}$ with leg 6 of $A^{[[4,5,6]]}$ produces a new tensor enumerator with 3 legs and 64 terms. Tracing the result of this with the tensor enumerator of the right Bacon-Shor code again produces a tensor enumerator with 3 legs and 64 terms.

When we trace in the perfect tensor on the left (or right) of lower wedge we will expand to four legs across the cut, and therefore will need to store a tensor enumerator with 256 terms. But this is the largest cut needed and so the enumerator of the whole code can be computing by manipulating tensors consisting of 256 enumerators. In some sense, this is optimal as tensor enumerator of the middle Bacon-Shor code has 4 legs itself, albeit only eight of its 256 terms are nonzero.

Example 45: Recall that the surface code can be built from contracting encoding isometries of the $[[5,1,2]]$ codes [31, §3]. Its encoding map has a non-trivial kernel and can be represented graphically by Figure 2 (left). We take the upward pointing legs as inputs while the downward pointing ones as outputs (physical qubits). The blue and red rank-1 tensors are $|+\rangle$ and $|0\rangle$ respectively. Note that [31] shows that the symmetries of this tensor network can be efficiently tracked and it uniquely fixes the target stabilizer code.

By taking each building block on the left diagram, we may construct its corresponding tensor enumerator, represented by a blue tensor node in Figure 2 (right). The network contraction yields the overall scalar enumerator of the $[[25,1,4]]$ surface code.

The examples we construct so far have a small number qubits such that their enumerators can also be computed by brute force. However, the tensor network method is far more scalable [41]. For example, the surface code tensor network may be easily extended to larger sizes as in Figure 3, which plots the coefficients of the double enumerator of a 3-by-150 surface code. $d_x, d_z$ are the $X$ and $Z$ weights respectively.
of this code has 748 qubits, brute force methods to compute the enumerator would involve checking the weights of at least $2^{747}$ terms, which is clearly infeasible. Yet as a tensor network the largest cut-distance one encounters is four. Hence the entire enumerator can be computed with less than a thousand $256 \times 256$ matrix multiplication (with polynomial entries).

VIII. CONCLUSION

In [31, Appendices D-E], it was shown that the stabilizer of the tensor trace of two stabilizer codes can efficiently be computed. However, as noted there, this operation can create nontrivial relationships among the logical degrees of freedom and hence there is no general way to predict the distance of the code after the trace. In this paper we partially addressed this by providing a mechanism for computing a quantum weight enumerator of the resulting code derived from those of the constituent building blocks. However these enumerators are themselves tensors, and the resulting enumerator is analogous to the tensor trace of the tensor enumerators of the building blocks.

The tensor enumerators we have defined satisfy many of the expected properties of a quantum enumerator. We can “trace” a high degree tensor enumerator to one of lower degree; and if traced down to a scalar we obtain the usual quantum weight enumerators. Under local unitary transformation, the tensor indices transform under an adjoint representation, while the other factors in the enumerator are invariant. Tensor enumerators are well-behaved under tensor product, in that the tensor enumerator of the tensor product of two codes is the tensor product of the tensor enumerators of each code. And, there is an analogue of the quantum MacWilliams transform that connects the two types of tensor enumerator. In developing this tensor enumerator formalism, we paved the way for applying tensor network methods to greatly improve the efficiency of computing quantum weight enumerator polynomials in practice. Although detailed applications of this formalism, including examples of larger quantum error correcting codes, are given in [41], we provided a few simple clarifying examples in this work. We further demonstrated the feasibility in obtaining the weight enumerator of a large rectangular surface code, which was impossible using other existing methods.

REFERENCES

[1] A. Y. Kitaev, “Quantum error correction with imperfect gates,” in Quantum Communication, Computing, and Measurement. Berlin, Germany: Springer, 1997, pp. 181–188.
[2] S. B. Bravyi and A. Y. Kitaev, “Quantum codes on a lattice with boundary,” 1998, arXiv:9811052.
[3] E. Dennis, A. Kitaev, A. Landahl, and J. Preskill, “Topological quantum memory,” J. Math. Phys., vol. 43, no. 9, pp. 4452–4505, Sep. 2002.
[4] J. Haah, “Local stabilizer codes in three dimensions without string logical operators,” Phys. Rev. A, Gen. Phys., vol. 83, no. 4, Apr. 2011, Art. no. 042330.
[5] A. Kitaev, “Anyons in an exactly solved model and beyond,” Ann. Phys., vol. 321, no. 1, pp. 2–111, Jan. 2006.
[6] H. Bombin, “Topological order with a twist: Ising anyons from an Abelian model,” Phys. Rev. Lett., vol. 105, no. 3, Jul. 2010, Art. no. 030403.
[7] M. A. Levin and X.-G. Wen, “String-net condensation: A physical mechanism for topological phases,” Phys. Rev. B, Condens. Matter, vol. 71, no. 4, Jan. 2005, Art. no. 045110.
[8] H. Bombin, “An introduction to topological quantum codes,” 2013, arXiv:1311.0277.
[9] R. Raussendorf and J. Harrington, “Fault-tolerant quantum computation with high threshold in two dimensions,” Phys. Rev. Lett., vol. 98, no. 19, May 2007, Art. no. 190504.
[10] A. G. Fowler, A. M. Stephens, and P. Groszkwoski, “High-threshold universal quantum computation on the surface code,” Phys. Rev. A, Gen. Phys., vol. 80, no. 5, Nov. 2009, Art. no. 052312.
[11] M. B. Hastings and A. Geller, “Reduced space-time and time costs using dislocation codes and arbitrary ancillas,” 2014, arXiv:1408.1379.
[12] S. Nagayama, A. G. Fowler, D. Horsman, S. J. Devitt, and R. V. Meter, “Surface code error correction on a defective lattice,” New J. Phys., vol. 19, no. 2, Feb. 2017, Art. no. 023050.
[13] R. Orús, “A practical introduction to tensor networks: Matrix product states and projected entangled pair states,” Ann. Phys., vol. 349, pp. 117–158, Oct. 2014.
[14] R. Orús, “Tensor networks for complex quantum systems,” Nature Rev. Phys., vol. 1, no. 9, pp. 358–550, Aug. 2019.
[15] A. J. Ferris and D. Poulin, “Tensor networks and quantum error correction,” Phys. Rev. Lett., vol. 113, no. 3, Jul. 2014, Art. no. 030501.
[16] F. Pastawski, B. Yoshida, D. Harlow, and J. Preskill, “Holographic quantum error-correcting codes: Toy models for the bulk/boundary correspondence,” J. High Energy Phys., vol. 2015, no. 6, p. 149, Jun. 2015.
[17] Z. Yang, P. Hayden, and X.-L. Qi, “Bidirectional holographic codes and sub-AdS locality,” J. High Energy Phys., vol. 2016, no. 1, p. 175, Jan. 2016.
[18] P. Hayden, S. Nezami, X.-L. Qi, N. Thomas, M. Walter, and Z. Yang, “Holographic duality from random tensor networks,” J. High Energy Phys., vol. 2016, no. 11, pp. 1–9, Nov. 2016.
[19] C. Cao and S. M. Carroll, “Bulk entanglement gravity without a boundary: Towards finding Einstein’s equation in Hilbert space,” Phys. Rev. D, Part. Fields, vol. 97, no. 8, Apr. 2018, Art. no. 082001.
[20] W. Donnelly, D. Marolf, B. Michel, and I. Wien, “Living on the edge: A toy model for holographic reconstruction of algebras with centers,” J. High Energy Phys., vol. 2017, no. 4, p. 93, Apr. 2017.
[21] C. Cao and B. Lackey, “Approximate Bacon-Shor code and holography,” J. High Energy Phys., vol. 2021, no. 5, p. 127, May 2021.
[22] R. J. Harris, N. A. McMahon, G. K. Brennen, and T. M. Stace, “Calderbank-Shor-Steane holographic quantum error-correcting codes,” Phys. Rev. A, Gen. Phys., vol. 98, no. 5, Nov. 2018, Art. no. 052301.
[23] R. J. Harris, E. Coupe, N. A. McMahon, G. K. Brennen, and T. M. Stace, “Decoding holographic codes with an integer optimization decoder,” Phys. Rev. A, Gen. Phys., vol. 102, no. 6, Dec. 2020, Art. no. 062417.
[24] A. Jahn, M. Gluza, F. Pastawski, and J. Eisert, “Holography and criticality in matchgate tensor networks,” Sci. Adv., vol. 5, no. 8, Aug. 2019, Art. no. eaaw0092.
[25] A. Jahn, M. Gluza, F. Pastawski, and J. Eisert, “Majorana dimers and holographic quantum error-correcting codes,” Phys. Rev. Res., vol. 1, no. 3, Nov. 2019, Art. no. 033079.
[26] A. Jahn, Z. Zimbórs, and J. Eisert, “Central charges of aperiodic holographic tensor-network models,” Phys. Rev. A, Gen. Phys., vol. 102, no. 4, Oct. 2020, Art. no. 042407.
[27] C. Cao, J. Pollack, and Y. Wang, “Hyperinvariant multiscale entanglement renormalization ansatz: Approximate holographic error correction codes with power-law correlators,” Phys. Rev. D, Part. Fields, vol. 105, no. 2, Jan. 2022, Art. no. 026018.
[28] A. Jahn, Z. Zimbórs, and J. Eisert, “Tensor network models of AdS/QaCFT,” Quantum, vol. 6, p. 643, Feb. 2022.
[29] S. Cree, K. Dolev, V. Calvera, and D. J. Williamson, “Fault-tolerant logical gates in holographic stabilizer codes are severely restricted,” PRX Quantum, vol. 2, no. 3, Aug. 2021, Art. no. 030337.
[30] A. Jahn and J. Eisert, “Holographic tensor network models and quantum error correction: A topical review,” Quantum Sci. Technol., vol. 6, no. 3, Jul. 2021, Art. no. 033002.
[31] C. Cao and B. Lackey, “Quantum Lego: Building quantum error correction codes from tensor networks,” PRX Quantum, vol. 3, no. 2, May 2022, Art. no. 020332.
[32] T. Farrelly, D. K. Tuckett, and T. M. Stace, “Local tensor-network codes,” New J. Phys., vol. 24, no. 4, Apr. 2022, Art. no. 043015.
[33] V. Paul Su, C. Cao, H.-Y. Hu, Y. Yanay, C. Tahan, and B. Swingle, “Discovery of optimal quantum error correcting codes via reinforcement learning,” 2023, arXiv:2305.06378.
[34] C. Mauron, T. Farrelly, and T. M. Stace, “Optimization of tensor network codes with reinforcement learning,” 2023, arXiv:2305.11470.
[35] P. Shor and R. Laflamme, “Quantum analog of the MacWilliams identities for classical coding theory,” Phys. Rev. Lett., vol. 78, no. 8, pp. 1600–1602, Feb. 1997.
[36] A. E. Ashikhmin, A. M. Barg, E. Knill, and S. N. Litsyn, “Quantum error detection. I. Statement of the problem,” IEEE Trans. Inf. Theory, vol. 46, no. 3, pp. 778–788, May 2000.

[37] E. M. Rains, “Quantum weight enumerators,” IEEE Trans. Inf. Theory, vol. 44, no. 4, pp. 1388–1394, Jul. 1998.

[38] E. M. Rains, “Quantum shadow enumerators,” IEEE Trans. Inf. Theory, vol. 45, no. 4, pp. 1388–1394, Jul. 1998.

[39] C. Hu, S. Yang, and S. S.-T. Yau, “Complete weight distributions and MacWilliams identities for asymmetric quantum codes,” IEEE Access, vol. 7, pp. 68404–68414, 2019.

[40] C. Hu, S. Yang, and S. S.-T. Yau, “Weight enumerators for nonbinary asymmetric quantum codes and their applications,” Adv. Appl. Math., vol. 121, Oct. 2020, Art. no. 102085.

[41] C. Cao, M. J. Gullans, B. Lackey, and Z. Wang, “Quantum Lego expansion pack: Enumerators from tensor networks,” 2023, arXiv:2308.05152.

[42] E. Knill, “Group representations, error bases and quantum codes,” 1996, arXiv:9608049.

[43] Z. Li and L. Xing, “Universal framework for quantum error-correcting codes,” Entropy, vol. 23, no. 8, p. 937, Jul. 2021.

[44] F. W. Werner, “All teleportation and dense coding schemes,” J. Phys. A, Math. Gen., vol. 34, no. 35, pp. 7081–7093, Sep. 2001.

[45] E. Knill, “Non-binary unitary error bases and quantum codes,” 1996, arXiv:9608048.

[46] A. A. Klappenecker and M. Rötteler, “Beyond stabilizer codes. I. Nice error bases,” IEEE Trans. Inf. Theory, vol. 48, no. 8, pp. 2392–2395, Aug. 2002.

[47] J. Macwilliams, “A theorem on the distribution of weights in a systematic code,” Bell Syst. Tech. J., vol. 42, no. 1, pp. 79–94, Jan. 1963.

[48] F. J. MacWilliams and N. J. A. Sloane, The Theory of Error Correcting Codes, vol. 16. Amsterdam, The Netherlands: Elsevier, 1977.

[49] A. Ashikhmin and E. Knill, “Nonbinary quantum stabilizer codes,” IEEE Trans. Inf. Theory, vol. 47, no. 7, pp. 3065–3072, Nov. 2001.

[50] L. Ioffe and M. Mézard, “Asymmetric quantum error-correcting codes,” Phys. Rev. A, Gen. Phys., vol. 75, no. 3, Mar. 2007, Art. no. 032345.

[51] P. K. Sarvepalli, A. Klappenecker, and M. Rötteler, “Asymmetric quantum LDPC codes,” in Proc. IEEE Int. Symp. Inf. Theory, Jul. 2008, pp. 305–309.

[52] A. M. Stephens, Z. W. E. Evans, S. J. Devitt, and L. C. L. Hollenberg, “Asymmetric quantum error correction via code conversion,” Phys. Rev. A, Gen. Phys., vol. 77, no. 6, Jun. 2008, Art. no. 062335.

[53] T. A. Springer, Invariant Theory, vol. 585. Berlin, Germany: Springer, 2006.

[54] B. Sturmfels, Algorithms in Invariant Theory. Berlin, Germany: Springer, 2008.

[55] The Sage Developers. (2022). SageMath, the Sage Mathematics Software System (Version 9.5). [Online]. Available: https://www.sagemath.org

[56] S. Bravyi and B. Terhal, “A no-go theorem for a two-dimensional self-correcting quantum memory based on stabilizer codes,” New J. Phys., vol. 11, no. 4, Apr. 2009, Art. no. 043029.

[57] W. K. Wootters, “A Wigner-function formulation of finite-state quantum mechanics,” Ann. Phys., vol. 176, no. 1, pp. 1–21, 1987.

[58] Z. Cai, J. Lu, and K. Stubbs, “On discrete Wigner transforms,” 2018, arXiv:1802.05834.

ChunJun Cao received the joint B.S. degree (Hons.) in mathematics and physics from The University of British Columbia in 2013 and the Ph.D. degree in physics from the California Institute of Technology (Caltech) in 2018. He was a Hartree Post-Doctoral Fellow with the Joint Center for Quantum Information and Computer Science, The University of Maryland, from 2018 to 2022; and a Post-Doctoral Scholar and a Visiting Associate with the Institute for Quantum Information and Matter, Caltech, from 2022 to 2023. He is currently an Assistant Professor with the Department of Physics, Virginia Tech. His research interests include quantum information theory, quantum gravity, and tensor networks.

Brad Lackey received the B.S., M.S., and Ph.D. degrees in mathematics from the University of Houston, Houston, TX, USA, in 1992 and 1996, respectively. He was an Izaak W. Killiam Memorial Post-Doctoral Fellow with the University of Alberta and an EPSRC “Fast-Track” Post-Doctoral Fellow with the University of Hull. From 2002 to 2018, he was an Applied Mathematician with the U.S. Department of Defense. Since 2018, he has been a Quantum Architect with Microsoft Corporation, Redmond, WA, USA. His research interests include quantum algorithms, quantum information theory, quantum programming languages, and quantum and post-quantum cryptography.