Quantum Persistent Homology

Bernardo Ameneyro[1], Vasileios Maroulas[1*], George Siopsis[2]

[1*]Department of Mathematics, The University of Tennessee, .
[2]Department of Physics and Astronomy, The University of Tennessee, .

*Corresponding author(s). E-mail(s): vasileios.maroulas@utk.edu;

Abstract

Persistent homology is a powerful mathematical tool that summarizes useful information about the shape of data allowing one to detect persistent topological features while one adjusts the resolution. However, the computation of such topological features is often a rather formidable task necessitating the sub-sampling the underlying data. To remedy this, we develop an efficient quantum computation of persistent Betti numbers, which track topological features of data across different scales. Our approach employs a persistent Dirac operator whose spectrum relates to that of the persistent combinatorial Laplacian, and thus allows us to recover the persistent Betti numbers which capture the persistent features of data. In addition, our algorithm can also extract the non-harmonic spectra of the Laplacian, which can be used for data analysis as well. We also test our algorithm on a point cloud data.

Keywords: topological data analysis, quantum persistent homology, persistent Dirac operator, persistent combinatorial Laplacian

1 Introduction

In recent years, topology and geometry blended with statistical methods have seen increasing application to the study of data analysis, visualization, and dimensionality reduction [1–9]. These applications range from classification and clustering in fields such as action recognition [10], handwriting analysis [11], and biology [12–16], to classification of high entropy alloy [17] and gas separation [18], to the analysis of complex biological networks [19], and other complex dynamical systems [20, 21] and sensor networks [22–26].

Persistent homology [27–31], the workhorse of topological data analysis (TDA), has helped to compress nonlinear point cloud data from a new geometrically faithful point of view. In the realm of signal analysis, classification and clustering based on topological features of the signal identifies features that traditional signal analysis techniques fail to detect [10, 12, 32–34]. Topological features, such as the number of connected components, cycles, and higher-dimensional voids, respectively, represent multi-stability, periodicity, and chaos in a dynamical system [12, 35].

The effective computation of persistence diagrams has become an area of intense active research, including a significant successful effort toward facilitating previously challenging computations. For example, the creation of persistence diagrams with packages such as Dionysus [36] and Ripser [37] take advantage of certain properties of simplicial complexes [38]. Point cloud data typically consist of many points. A set of n such points possesses 2^n potential subsets that could contribute to the topology. Classical algorithms for persistent homology (e.g., see [39]) typically need $\mathcal{O}(2^{3n})$ operations to diagonalize a $2^n \times 2^n$ boundary matrix and obtain the topological information. For this reason, classical implementations often restrict attention to features of lower dimensions.

More recently, quantum algorithms that compute topological features of data were developed. The first such algorithm was introduced in [40] with run time $O(n^5/\alpha^2)$, which seems to be exponentially faster than the best known classical algorithms. This algorithm was designed for a discrete-variable (DV) quantum system based on qubits, and it was extended to a continuous-variable (CV) substrate in [41]. These two algorithms compute Betti numbers by analyzing a linear operator called the Dirac operator whose square is the combinatorial Laplacian. However, the study did not address any real implementation on data. To that end, a few recent attempts on data for tracking the non-persistent topological features were studied. Indeed, the work in [42] demonstrates a quantum algorithm presented in [40] by employing a six-photon quantum processor to successfully analyze the topological features of a network including three data points. Moreover the work in [43] developed a quantum annealing approach for topologically analyzing point cloud data, and [44] gives a quantum circuit to construct all maximal cliques of an n-node network using Grover's search algorithm. Further studies [45–47] propose new implementations that attempt to avoid the quantum memory and fault tolerance requirements of [40]. However, as it was noted in [48], these algorithms do not compute persistent topological features so that one may track how topological features persist as the resolution of data changes and the underlying noise may vary. The works in [49, 50] tackle this issue using quantum singular value transformation on a unitary block encoding of the projection operators onto the kernels of the restricted boundary and coboundary operators that appear in the persistent combinatorial (or Hodge) Laplacian, see [51, 52].

The contribution of our manuscript is as follows. We introduce a persistent Dirac operator similar to the Dirac operator presented in [40], with the key difference being that ours is able to track features across different scales. We show that the eigenvalues and eigenspaces of a shifted version of this persistent Dirac operator correspond to the spectrum of the persistent combinatorial Laplacian, and we provide a quantum algorithm that relies on Grover's search and quantum phase estimation to recover the dimensions of these eigenspaces. In contrast to other quantum algorithms for persistent homology [49, 50], ours not only

obtains the persistent Betti numbers, but also the non-harmonic spectra of the persistent combinatorial Laplacian, which can also be used for data analysis tasks, e.g. see [51, 53, 54].

Our paper is organized as follows. In Section 2, we review persistent homology by introducing the important concepts relevant to the algorithm, such as the persistent combinatorial Laplacian and persistent Betti numbers. In Section 3, we introduce the persistent Dirac operator and show its relationship with the persistent combinatorial Laplacian, we also present an outline of the quantum algorithm to recover the spectra of the persistent Dirac operator and detail our main contributions. In Section 4, we apply our algorithm to a data set whose points are organized in squares, which was suggested in [48] as a case in which persistent Betti numbers differ from Betti numbers. Finally, we offer concluding remarks in Section 5. Details of subroutines needed for our quantum algorithm are provided in Appendices A, B, and C (Membership oracle, Grover's search, and implementation of an exponential operator, respectively).

2 Persistent Homology

Persistent homology studies objects called simplicial complexes. The classical algorithms that perform topological data analysis use the data to construct simplicial complexes, e.g., by connecting all the points in a point cloud within a certain distance from each other (Vietoris-Rips complex) and then varying the distance to obtain a filtration of simplicial complexes. After that, the algorithms proceed to compute the eigenvalues and eigenvectors of linear operators, such as the boundary operator and the persistent combinatorial Laplacian that act on the complexes.

We start by defining simplicial complexes and homology, an algebraic descriptor for coarse shape in topological spaces, and in turn persistent homology, which harnesses the power of homology to the description of subspace filtrations of topological spaces.

Definition 1. *A κ-dimensional manifold X is a topological space such that if every point x that belongs to X has a homeomorphic neighborhood to an open neighborhood in the κ-dimensional Euclidean space.*

Next, we expand the definition of the κ-dimensional manifold to κ-simplex which may be topologically treated as a κ-dimensional manifold including its boundary.

Definition 2. *A κ-simplex is defined by $\kappa+1$ linearly independent vertices as the collection of all their convex combinations:*

$$\sigma = [v_0, ..., v_\kappa] = \left\{ \sum_{i=0}^{\kappa} \alpha_i v_i : \sum_{i=0}^{\kappa} \alpha_i = 1 \text{ and } \alpha_i \geq 0 \text{ for all } i \right\}. \tag{1}$$

An oriented simplex is expressed as the ordered list of its vertices, such as $[v_0, v_1, v_2]$. The faces of a simplex consist of all the simplices generated by a subset of its vertex set.

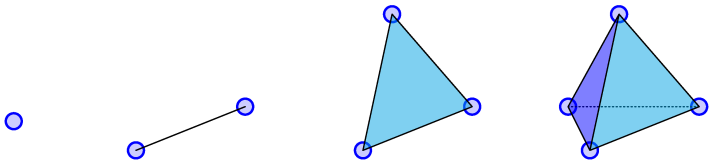

Fig. 1: Simplices of dimension 0 through 3.

Fig. 1 shows the simplices of dimensions zero (vertex), one (edge), two (triangle), and three (tetrahedron).

Definition 3. *A simplicial complex $\mathcal{K}$ is a collection of simplices satisfying the following*

(i) if $\sigma \in \mathcal{K}$, then all its faces $\tau \subset \sigma$ are also in $\mathcal{K}$, and

(ii) the intersection $\sigma_1 \cap \sigma_2$ of any pair of simplices $\sigma_1, \sigma_2 \in \mathcal{K}$ is another simplex in $\mathcal{K}$.

The collection of κ-simplices within $\mathcal{K}$ is written here as $\mathcal{K}_\kappa$.

Furthermore, we can use these basis-like collections to define a space of formal sums of simplices. These spaces are called chain groups and are akin to vector spaces, an important feature since quantum states are elements of a Hilbert space over $\mathbb{C}$.

Definition 4. *Let's denote with $C_\kappa(\mathcal{K})$ the chain group of dimension κ on a simplicial complex $\mathcal{K}$, which is defined by*

$$C_\kappa(\mathcal{K}) = \left\{ \sum_{\sigma \in \mathcal{K}_\kappa} n_\sigma \sigma : n_\sigma \in \mathbb{C} \right\}. \tag{2}$$

Relying on Def. 4, one understands that chain groups give an algebraic way to describe subsets of simplices as a formal sum. For example, the boundary of a triangle (Fig. 2) is considered the sum of its three edges, and the boundary of an edge yields the sum of its endpoints. The presence of sign specifies the simplex's orientation. This notion of a boundary is fundamental to persistent homology and is formalized as a map between chain groups.

Definition 5. *The κ-th boundary map is a homomorphism on the chain groups $\partial_\kappa : C_\kappa(\mathcal{K}) \to C_{\kappa-1}(\mathcal{K})$ defined by its action on each simplex:*

$$\partial_\kappa [v_0, ..., v_\kappa] = \sum_{n=0}^{\kappa} (-1)^n [v_0, ..., v_{n-1}, v_{n+1}, ..., v_\kappa], \tag{3}$$

as an alternating sum over the faces of dimension $\kappa - 1$.

One may obtain a chain complex by taking into account chain groups (and their boundary maps) of any dimension

$$... \xrightarrow{\partial_{\kappa+1}} C_\kappa(\mathcal{K}) \xrightarrow{\partial_\kappa} C_{\kappa-1}(\mathcal{K}) \xrightarrow{\partial_{\kappa-1}} ... \xrightarrow{\partial_1} C_0(\mathcal{K}) \xrightarrow{\mathbf{0}} 0. \tag{4}$$

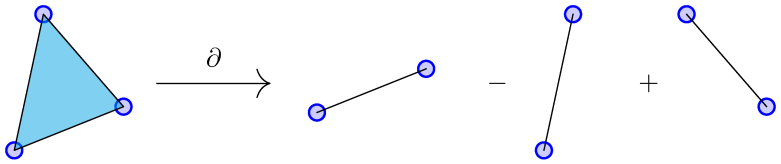

Fig. 2: Boundary of a triangle

As it can be shown, e.g., see in [3], the composition of subsequent boundary maps $\partial_{\kappa+1}\partial_\kappa = 0$, which yields that $\mathrm{im}(\partial_{\kappa+1}) \subset \ker(\partial_\kappa)$. The κ-Betti number, denoted by β_κ, is the dimension of the homology group $H_\kappa(\mathcal{K}) = \ker(\partial_\kappa)/\,\mathrm{im}(\partial_{\kappa+1})$.

The homology groups are generated by the topological features of the complex $\mathcal{K}$, i.e., generators for the 0-homology group correspond to connected components, generators of 1-homology group correspond to holes in $\mathcal{K}$, etc.

Next, when analyzing a discrete data set $\boldsymbol{x} = \{v_i\}_{i=1}^N$ that belongs to a metric space (X, d) it is not enough to consider the set itself as a simplicial complex for its homology would simply yield the number of data points in the set. Instead, we take advantage of the metric d to obtain more information. Indeed, we fix a radius $r > 0$ and consider the collection of neighborhoods (think of them as balls centered at each datum) $U = \{U_i\} = \{B(v_i, r)\}$ along with its union $\mathcal{U}_r = \cup_i B(v_i, r)$. The collection of sets $\{\mathcal{U}_r\}_{r\in\mathbb{R}^+}$ provides information about the arrangement of the dataset $\boldsymbol{x}$ at different scales r (see e.g. Fig. 3). To make homological computations tractable for $\mathcal{U}_r$, one may consider the Vietoris-Rips (VR) complexes.

Definition 6. *The Vietoris-Rips complex of the point cloud data $\boldsymbol{x} = \{v_i\}$ at scale ϵ, denoted by $S^\epsilon(\boldsymbol{x})$ (or simply S^ϵ), is the simplicial complex where a κ-simplex $\sigma = [v_{i_0}, ..., v_{i_\kappa}]$ is in $S^\epsilon(\boldsymbol{x})$ if and only if $diam(\sigma) \leq \epsilon$, where $diam(\sigma) := \max_{j,j'} \left\{ d(v_{i_j}, v_{i_{j'}}) \right\}$ denotes the diameter of the simplicial complex S^ϵ defined as the maximal distance between $v_{i_j}, v_{i_{j'}} \in S^\epsilon$.*

Unfortunately the tools defined so far would only allow to obtain the Betti numbers at a fixed scale ϵ. However data analysis often requires to consider them at different scales so that we verify which topological features persist for various ϵ as in Def. 6, while those that persist for a short time may be just noise. So, we must extend the notions above to include multiple simplicial complexes at once.

Consider a nested sequence or filtration of simplicial complexes

$$\emptyset \subseteq \mathcal{K}_1 \subseteq \mathcal{K}_2 \subseteq ... \subseteq \mathcal{K}_n. \tag{5}$$

Given a point cloud data set $\boldsymbol{x}$, one can construct a filtration of VR complexes $\emptyset \subseteq S^{\epsilon_1} \subseteq S^{\epsilon_2} \subseteq ... \subseteq S^{\epsilon_N} = S$ by choosing an increasing sequence of scales $0 < \epsilon_1 \leq \epsilon_2 \leq \ldots \epsilon_N$, see Figure 3 for an example. Indeed, it follows from Def. 6 that if $\epsilon \leq \epsilon'$ then $S^\epsilon \subseteq S^{\epsilon'}$. Moreover, one can check that there is a

maximum of 2^n simplices that can be built from a data set with n points. So, there is a maximal VR complex S of size 2^n that contains all VR complexes S^ϵ.

Each complex $\mathcal{K}_i$ in Eq. (5) has associated chain groups $C_\kappa(\mathcal{K}_i)$, boundary operators ∂_κ^i and Homology groups H_κ^i as before. But now the inclusion maps $\iota : \mathcal{K}_i \to \mathcal{K}_j$ between complexes induce homomorphisms $h_\kappa^{i,j} = \iota_* : C_\kappa(\mathcal{K}_i) \to C_\kappa(\mathcal{K}_j)$ between their corresponding chain groups.

Definition 7. *For two nested simplicial complexes $\mathcal{K}_i \subseteq \mathcal{K}_j$ we define their κ-th persistent Homology group $H_\kappa^{i,j}$ as*

$$H_\kappa^{i,j} = Im(h_\kappa^{i,j}) = Ker(\partial_\kappa^i)/(Im(\partial_{\kappa+1}^j) \cap Ker(\partial_\kappa^i)) \tag{6}$$

where $Ker(\partial_\kappa^i)$ is viewed as a subgroup of $Ker(\partial_\kappa^j)$. The dimension of $H_\kappa^{i,j}$ is the κ-th persistent Betti number $\beta_\kappa^{i,j}$.

Let $\tilde{C}_{\kappa+1}^{i,j} = \left\{x \in C_{\kappa+1}(\mathcal{K}_j) : \partial_{\kappa+1}^j x \in C_\kappa(\mathcal{K}_i)\right\}$, that is, the subgroup of $C_{\kappa+1}(\mathcal{K}_j)$ defined by the $(\kappa+1)$-simplices in $\mathcal{K}_j$ with boundary in $\mathcal{K}_i$. Then we can define the κ-th persistent combinatorial Laplacian

$$\mathcal{L}_\kappa^{i,j} = \tilde{\partial}_\kappa^{i,i*}\tilde{\partial}_\kappa^{i,i} + \tilde{\partial}_{\kappa+1}^{i,j}\tilde{\partial}_{\kappa+1}^{i,j*}, \tag{7}$$

where $\tilde{\partial}_{\kappa+1}^{i,j}$ is the restriction of the boundary operator $\partial_{\kappa+1}^j$ to $\tilde{C}_{\kappa+1}^{i,j}$. Using [51], one may show that the dimension of the kernel of $\mathcal{L}_\kappa^{i,j}$ is the κ-th persistent Betti number $\beta_\kappa^{i,j}$.

3 Quantum algorithm

Our quantum algorithm proceeds by first mapping the simplices and other concepts from persistent homology onto quantum states and Hermitian operators. We choose a direct mapping between the n vertices of a point cloud and n qubits. In this manner, all 2^n possible simplices can be mapped onto quantum computational basis states in an n-qubit Hilbert space, using exponentially fewer memory space than the classical case. This allows us to use the fermionic representation of the boundary operator [45, 55], which is defined in terms of Pauli operators. Moreover, the chain groups are identified by closed subspaces of the n-qubit Hilbert space, so we can use their respective orthogonal projection to restrict the boundary operator as in Eq. (7) for the persistent combinatorial Laplacian. To implement these projections we suggest the use of a quantum memory along with Grover's search, however NISQ friendly alternatives have been discussed in other works [45–47]. Finally, our algorithm uses quantum phase estimation to recover the spectra of the persistent combinatorial Laplacian, including the persistent Betti numbers. Our main contributions include the implementation of a projection onto the closed subspace that encodes the chain group $C^{\epsilon,\epsilon'}$. In addition, we introduce the persistent Dirac operator, which allows our algorithm to recover the full spectrum of the persistent combinatorial Laplacian, in contrast to other quantum algorithms for persistent homology [49, 50] that can only obtain the kernel.

3.1 Representing simplices and chain complexes with quantum states

A k-simplex $\sigma = [v_{i_1}, \ldots, v_{i_k}]$ can be stored in a n-qubit register as $|\sigma\rangle = |v_1\rangle \otimes \cdots \otimes |v_n\rangle$, with 1s at the positions of its $k+1$ vertices $v_{i_1}, \ldots, v_{i_k}$ and 0s elsewhere. Let S denote the collection of all possible quantum basis states of n qubits given by a string of n 0s and 1s, then S effectively encodes the maximal VR complex described in Section 2 with all possible simplices that can be formed with n vertices or data points. Moreover, $\mathcal{H}$, the Hilbert space over $\mathbb{C}$ with basis S encodes the chain group of that complex.

We denote by S_k the subset of states in S encoding k-simplices, and by $\mathcal{H}_k$ the corresponding closed subspace of $\mathcal{H}$ which encodes the k-th chain group defined in Eq. (2). Notice that the dimension of a simplex can be recovered from its quantum state encoding by counting the number of qubits in state 1.

Similarly, we write S^ϵ for the subset of S encoding simplices of diameter at most ϵ, in other words the VR complex at scale ϵ as in Def. 6, and $\mathcal{H}^\epsilon$ for the corresponding closed subspace of $\mathcal{H}$. We also need to consider the closed subspace of $\mathcal{H}$ that encodes the chain group $\tilde{C}^{\epsilon,\epsilon'}$ of elements present at scale ϵ' with boundary in scale ϵ. This subspace is given by

$$\mathcal{H}^{\epsilon,\epsilon'} = \{|\psi\rangle \in \mathcal{H}^{\epsilon'} : \partial |\psi\rangle \in \mathcal{H}^{\epsilon}\}. \tag{8}$$

A boundary operator can be defined on $\mathcal{H}$ using Pauli gates X, Y, Z as

$$\partial := \sum_{i=0}^{n-1} Z^{\otimes(n-1-i)} \otimes X^+ \otimes I^{\otimes i} \ , \tag{9}$$

where $X^\pm = \frac{1}{2}(X \pm iY)$. Since $X^+ |1\rangle = |0\rangle$ and $X^+ |0\rangle = 0$, it maps $\mathcal{H}_k$ to $\mathcal{H}_{k-1}$. This representation of the boundary map introduced in [45] has various properties. In particular, notice that ∂ is a bounded continuous linear operator on $\mathcal{H}$ such that $\partial^2 = 0$. The reader may refer to [55] for further details on these properties, and its advantages over the representation introduced in [40]. Mapping $\mathcal{H}_{k-1}$ to $\mathcal{H}_k$, its adjoint is given by $\partial^* := \sum_{i=0}^{n-1} Z^{\otimes(n-1-i)} \otimes X^- \otimes I^{\otimes i}$. In order to obtain a Hermitian version of the boundary operator in Eq. (9) that can act on the subspaces considered by the persistent combinatorial Laplacian in Eq. (7), we attach two ancillary qubits and define the Dirac operator

$$B = (|0\rangle\langle 1| + |1\rangle\langle 2|) \otimes \partial + (|1\rangle\langle 0| + |2\rangle\langle 1|) \otimes \partial^* \tag{10}$$

on the Hilbert space spanned by states $\{|0\rangle, |1\rangle, |2\rangle\}$, where $|0\rangle = |0\rangle \otimes |0\rangle$, $|1\rangle = |0\rangle \otimes |1\rangle$, and $|2\rangle = |1\rangle \otimes |0\rangle$.

3.2 Projections

To implement a projection P_k onto the subspace $\mathcal{H}_k$ generated by simplices of dimension k it suffices to count the number of qubits in state 1. Indeed, computational basis states in $\mathcal{H}_k$ are exactly those that have $k+1$ qubits in state 1. The reader may refer to [45, 47] for details on efficient implementations.

On the other hand, for a scale ϵ, we assume the existence of an oracle

$$\mathcal{O}^\epsilon|\sigma\rangle|0\rangle = |\sigma\rangle|a^\epsilon_\sigma\rangle \ , \tag{11}$$

where $a^\epsilon_\sigma = 1_{\sigma\in S^\epsilon}$ determines the membership in S^ϵ, which in turn encodes the simplices in the ϵ-complex. For completeness we provide a construction of such oracle in Appendix A.

The oracle in Eq. (11) can be used for the implementation of the projection operators,

$$P^\epsilon = \sum_{\sigma\in S^\epsilon} |\sigma\rangle\langle\sigma|, \tag{12}$$

onto $\mathcal{H}^\epsilon$, the subspace of $\mathcal{H}$ spanned by the simplices in S^ϵ, which will be needed to construct the persistent Dirac operator later in Section 3.3. To implement the projection P^ϵ, we perform amplitude amplification [56] based on Grover's search algorithm [57] (for details, see Appendix B).

For a NISQ implementation of the projections P^ϵ, i.e. an implementation that doesn't require a fault tolerant computer like Grover's search algorithm or a quantum memory, see the approach by Ubaru et. al. [45]. This approach involves classical computation of the pairwise distance of the vertices as well as the $n\times n$ ϵ-adjacency matrix $\{A_{i,j}\}$. The quantum circuit is then built by adding gates conditional to the values $A_{i,j}$.

To implement the projection $P^{\epsilon,\epsilon'}$ onto the subspace $\mathcal{H}^{\epsilon,\epsilon'}$ defined in Eq. (8), we introduce the operator $\mathcal{W} = (|0\rangle\langle 1| + |1\rangle\langle 0|)\otimes I - |0\rangle\langle 2|\otimes\partial - |2\rangle\langle 0|\otimes\partial^*$, defined with the aid of a pair of qubits, along with the projection $Q^{\epsilon,\epsilon'} = |0\rangle\langle 0|\otimes I + |1\rangle\langle 1|\otimes P^\epsilon + |2\rangle\langle 2|\otimes P^{\epsilon'}$. Then, we consider, $\mathcal{W}^{\epsilon,\epsilon'} = Q^{\epsilon,\epsilon'}(I-\mathcal{W})Q^{\epsilon,\epsilon'}$, acting on a vector of the form $|\phi\rangle = |0\rangle|\phi_0\rangle + |1\rangle|\phi_1\rangle + |2\rangle|\phi_2\rangle$. Notice that $\mathcal{W}^{\epsilon,\epsilon'}$ acts as the identity on states that satisfy

$$\partial P^{\epsilon'}|\phi_2\rangle = P^\epsilon|\phi_1\rangle \tag{13a}$$

$$(I-P^\epsilon)|\phi_1\rangle = P^\epsilon|\phi_0\rangle \tag{13b}$$

$$(I-P^{\epsilon'})|\phi_2\rangle = P^{\epsilon'}\partial^*|\phi_0\rangle\,. \tag{13c}$$

In particular, (13b) and (13c) only hold if both sides are equal to zero. Thus, we may conclude that $|\phi_1\rangle \in \mathcal{H}^{\epsilon}$ and $|\phi_2\rangle \in \mathcal{H}^{\epsilon'}$. It follows then from Eq. (13a) that $\partial |\phi_2\rangle \in \mathcal{H}^{\epsilon}$, so that $|\phi_2\rangle \in \mathcal{H}^{\epsilon,\epsilon'}$. Therefore, the projection operator $P^{\epsilon,\epsilon'}$ can be implemented by projecting onto the eigenspace of $\mathcal{W}^{\epsilon,\epsilon'}$ with eigenvalue 1.

3.3 Persistent Dirac operator

For persistent homology, we need to restrict the space on which the Dirac operator acts. Precisely, the Dirac operator should act on quantum states of the form $|0\rangle |\psi_0\rangle + |1\rangle |\psi_1\rangle + |2\rangle |\psi_2\rangle$, where $|\psi_0\rangle \in \mathcal{H}^{\epsilon}_{k-1}$, $|\psi_1\rangle \in \mathcal{H}^{\epsilon}_{k}$, and $|\psi_2\rangle \in \mathcal{H}^{\epsilon,\epsilon'}_{k+1}$. Moreover, we want the boundary operator and its adjoint to act like the restricted boundary and coboundary in Eq. (7) of the Laplacian. To that end, we introduce the persistent Dirac operator which plays a central role in our quantum algorithm.

Definition 8. *Let P^{ϵ} and $P^{\epsilon,\epsilon'}$ be the respective orthogonal projections onto the closed subspaces $\mathcal{H}^{\epsilon}$ and $\mathcal{H}^{\epsilon,\epsilon'}$ of $\mathcal{H}$, defined in Section 3.2. Then, the persistent Dirac operator $B^{\epsilon,\epsilon'}$ is defined as $B^{\epsilon,\epsilon'} = -PBP$, where B is the Dirac operator in Eq. (10) and*

$$P = (|0\rangle \langle 0| - |1\rangle \langle 1|) \otimes P^{\epsilon} + |2\rangle \langle 2| \otimes P^{\epsilon,\epsilon'}. \tag{14}$$

We can visualize this operator as a block matrix

$$B^{\epsilon,\epsilon'} = \begin{pmatrix} 0 & P^{\epsilon}\partial P^{\epsilon} & 0 \\ P^{\epsilon}\partial^* P^{\epsilon} & 0 & P^{\epsilon}\partial P^{\epsilon,\epsilon'} \\ 0 & P^{\epsilon,\epsilon'}\partial^* P^{\epsilon} & 0 \end{pmatrix} \tag{15}$$

The importance of the persistent Dirac operator is that we can use it to recover the eigenvalues of the persistent combinatorial Laplacian from Eq. (7). This is summarized below in Theorem 1.

Theorem 1. *The positive eigenspaces of the persistent combinatorial Laplacian introduced in Eq. (7) have a one-to-one correspondence with the eigenspaces of the persistent Dirac operator introduced in Def. 8 corresponding to positive eigenvalues. That is, $\lambda > 0$ is an eigenvalue of the persistent Dirac operator, defined in Eq. (15), with eigenvector $|0\rangle |\psi_0\rangle + |1\rangle |\psi_1\rangle + |2\rangle |\psi_2\rangle$ if and only if $|\psi_0\rangle, |\psi_1\rangle \in \mathcal{H}^{\epsilon}$ and $|\psi_2\rangle \in \mathcal{H}^{\epsilon,\epsilon'}$, and $|\psi_1\rangle$ is an eigenvector of the persistent combinatorial Laplacian $\mathcal{L}^{\epsilon,\epsilon'}$ from Eq. (7) with eigenvalue λ^2. Moreover, $|\psi_0\rangle$ and $|\psi_2\rangle$ are uniquely defined by $|\psi_1\rangle$.*

Proof. The vector $|0\rangle|\psi_0\rangle + |1\rangle|\psi_1\rangle + |2\rangle|\psi_2\rangle$ is an eigenstate of the persistent Dirac operator if and only if it satisfies the conditions

$$P^{\epsilon}\partial P^{\epsilon}|\psi_1\rangle = \lambda|\psi_0\rangle \tag{16a}$$

$$P^{\epsilon}\partial P^{\epsilon,\epsilon'}|\psi_2\rangle + P^{\epsilon}\partial^* P^{\epsilon}|\psi_0\rangle = \lambda|\psi_1\rangle \tag{16b}$$

$$P^{\epsilon,\epsilon'}\partial^* P^{\epsilon}|\psi_1\rangle = \lambda|\psi_2\rangle \tag{16c}$$

Notice that the left hand side of Equations (16a) and (16b) are contained in $\mathcal{H}^{\epsilon}$ because of the projections, implying that $|\psi_0\rangle, |\psi_1\rangle \in \mathcal{H}^{\epsilon}$. Similarly, Eq. (16c) is satisfied when $|\psi_2\rangle \in \mathcal{H}^{\epsilon,\epsilon'}$. Moreover, since $\lambda \neq 0$, Eqs. (16a) and (16c) express $|\psi_0\rangle$ and $|\psi_2\rangle$ in terms of $|\psi_1\rangle$, and substituting into Eq. (16b), we obtain $\left(P^{\epsilon}\partial^*\partial P^{\epsilon} + \partial P^{\epsilon,\epsilon'}\partial^* P^{\epsilon}\right)|\psi_1\rangle = \lambda^2|\psi_1\rangle$. □

However, the above theorem does not hold for $\lambda = 0$, because the states $|\psi_0\rangle$ and $|\psi_2\rangle$ are no longer uniquely defined by $|\psi_1\rangle$. In particular, the kernel of the persistent Dirac operator contains elements of $\mathcal{H}^{\epsilon\perp}$ and $\mathcal{H}^{\epsilon,\epsilon'\perp}$. To recover information about the kernel of the persistent combinatorial Laplacian, i.e., the persistent Betti numbers, we must introduce a gap in the spectrum by considering a shifted version of the persistent Dirac operator.

Definition 9. *Let $B^{\epsilon,\epsilon'}$ denote the persistent Dirac operator introduced in Def. 8, and take $\xi > 0$. Then we define the ξ-shift persistent Dirac operator $B^{\epsilon,\epsilon'}[\xi]$ as $B^{\epsilon,\epsilon'}[\xi] := B^{\epsilon,\epsilon'} - \xi P$, where P is defined in (14).*

Based on Definition 9, one may rewrite $B^{\epsilon,\epsilon'}[\xi]$ as a block matrix

$$B^{\epsilon,\epsilon'}[\xi] = \begin{pmatrix} -\xi P^{\epsilon} & P^{\epsilon}\partial P^{\epsilon} & 0 \\ P^{\epsilon}\partial^* P^{\epsilon} & \xi P^{\epsilon} & P^{\epsilon}\partial P^{\epsilon,\epsilon'} \\ 0 & P^{\epsilon,\epsilon'}\partial^* P^{\epsilon} & -\xi P^{\epsilon,\epsilon'} \end{pmatrix}. \tag{17}$$

Next, by shifting the persistent Dirac operator, we introduce a gap in its spectrum which is needed to avoid overcounting of vectors in the kernel of the persistent combinatorial Laplacian. Theorem 2 states that we can use the positive eigenvalues of the shifted persistent Dirac operator to recover the full spectrum of the persistent combinatorial Laplacian $\mathcal{L}^{\epsilon,\epsilon'}$.

Theorem 2. *The eigenspaces of the persistent combinatorial Laplacian introduced in Eq. (7) have a one-to-one correspondence with the eigenspaces of the ξ-shift persistent Dirac operator introduced in Def. 9 corresponding to a positive eigenvalue. In particular, for any eigenvalue γ of the persistent combinatorial Laplacian $\mathcal{L}^{\epsilon,\epsilon'}$ with corresponding eigenvector $|\psi_1\rangle$, there is a unique choice of $\lambda > 0$, $|\psi_0\rangle$ and $|\psi_2\rangle$*

such that the quantum state $|0\rangle|\psi_0\rangle + |1\rangle|\psi_1\rangle + |2\rangle|\psi_2\rangle$ *is an eigenvector of* $B^{\epsilon,\epsilon'}[\xi]$ *with eigenvalue* λ*. Moreover,* $|\psi_0\rangle, |\psi_1\rangle \in \mathcal{H}^{\epsilon}$ *and* $|\psi_2\rangle \in \mathcal{H}^{\epsilon,\epsilon'}$*, and the eigenvalues satisfy* $\lambda^2 - \xi^2 = \gamma$*.*

Proof. The vector $|0\rangle|\psi_0\rangle + |1\rangle|\psi_1\rangle + |2\rangle|\psi_2\rangle$ is an eigenstate of the ξ-shift persistent Dirac operator if and only if it satisfies the conditions

$$-\xi P^{\epsilon}|\psi_0\rangle + P^{\epsilon}\partial P^{\epsilon}|\psi_1\rangle = \lambda|\psi_0\rangle \tag{18a}$$

$$P^{\epsilon}\partial P^{\epsilon,\epsilon'}|\psi_2\rangle + \xi P^{\epsilon}|\psi_1\rangle + P^{\epsilon}\partial^* P^{\epsilon}|\psi_0\rangle = \lambda|\psi_1\rangle \tag{18b}$$

$$P^{\epsilon,\epsilon'}\partial^* P^{\epsilon}|\psi_1\rangle - \xi P^{\epsilon,\epsilon'}|\psi_2\rangle = \lambda|\psi_2\rangle \tag{18c}$$

Notice that for $\lambda > 0$ the left hand side of Eqs. (18a) and (18b) are again contained in $\mathcal{H}^{\epsilon}$, similarly the left side of Eq. (18c) is in $\mathcal{H}^{\epsilon,\epsilon'}$. This implies that the corresponding eigenvectors must satisfy $|\psi_0\rangle, |\psi_1\rangle \in \mathcal{H}^{\epsilon}$ and $|\psi_2\rangle \in \mathcal{H}^{\epsilon,\epsilon'}$. Therefore, we may rewrite the conditions as

$$P^{\epsilon}\partial P^{\epsilon}|\psi_1\rangle = (\lambda + \xi)|\psi_0\rangle \tag{19a}$$

$$P^{\epsilon}\partial P^{\epsilon,\epsilon'}|\psi_2\rangle + P^{\epsilon}\partial^* P^{\epsilon}|\psi_0\rangle = (\lambda - \xi)|\psi_1\rangle \tag{19b}$$

$$P^{\epsilon,\epsilon'}\partial^* P^{\epsilon}|\psi_1\rangle = (\lambda + \xi)|\psi_2\rangle \tag{19c}$$

Furthermore, since $\lambda \neq -\xi$, we can express $|\psi_0\rangle$ and $|\psi_2\rangle$ in terms of $|\psi_1\rangle$ using Equations (19a) and (19c), respectively, and then substitute into Eq. (19b) to obtain $\left(P^{\epsilon}\partial^*\partial P^{\epsilon} + \partial P^{\epsilon,\epsilon'}\partial^* P^{\epsilon}\right)|\psi_1\rangle = (\lambda^2 - \xi^2)|\psi_1\rangle$. Therefore, $|\psi_1\rangle$ is an eigenvector of the persistent combinatorial Laplacian $\mathcal{L}^{\epsilon,\epsilon'}$ with eigenvalue $\gamma = \lambda^2 - \xi^2$. In particular, the kernel of $\mathcal{L}^{\epsilon,\epsilon'}$ and hence the persistent Betti numbers are given by $\lambda = \xi$.

□

3.4 Quantum Phase Estimation

We may recover the persistent Betti number $\beta_k^{\epsilon,\epsilon'}$ as the dimension of the eigenspace of the ξ-shift persistent Dirac operator $B^{\epsilon,\epsilon'}[\xi]$ introduced in Def. 9 with eigenvalue $\lambda = \xi$. To estimate the dimensions of the eigenspaces of the shifted persistent Dirac operator we use a quantum phase estimation algorithm described below, see Figure 5 for an example quantum circuit. This method not only yields the persistent Betti numbers but also the non-harmonic spectra of the Laplacian in Eq. (7).

For explicit calculations, we restrict $B^{\epsilon,\epsilon'}[\xi]$ to the desired dimension k, by restricting attention to the Hilbert space of states $|\psi\rangle = |0\rangle|\psi_0\rangle + |1\rangle|\psi_1\rangle + |2\rangle|\psi_2\rangle$, where $|\psi_0\rangle \in \mathcal{H}_{k-1}$, $|\psi_1\rangle \in \mathcal{H}_k$, $|\psi_2\rangle \in \mathcal{H}_{k+1}$

(see Appendix C). Starting with the uniform state in Eq. (C3), we copy the basis states to create the maximally entangled state $|\tilde{\boldsymbol{s}}\rangle_{12} = \frac{1}{\sqrt{\boldsymbol{N}}} \sum_{a=1}^{\boldsymbol{N}} |e_a\rangle_1 |e_a\rangle_2$. Since $B_k^{\epsilon,\epsilon'}[\xi]$ is self-adjoint, we can find an orthonomral basis Λ of eigenvectors $|\lambda\rangle$ and rewrite $|\tilde{\boldsymbol{s}}\rangle_{12} = \frac{1}{\sqrt{\boldsymbol{N}}} \sum_{|\lambda\rangle\in\Lambda} |\lambda\rangle_1 |\lambda\rangle_2$.

Next, we introduce a register of qubits R in the state $|R\rangle_R = \frac{1}{\sqrt{M}} \sum_{y=0}^{M-1} |y\rangle_R$ and entangle the registers 2 and R by applying U_B, a unitary version of $B_k^{\epsilon,\epsilon'}[\xi]$ given by

$$U_B = e^{2\pi i l y B_k^{\epsilon,\epsilon'}[\xi]/M}, \tag{20}$$

where M and l are positive integers that can be adjusted at will. We obtain the quantum state $U_B |\tilde{\boldsymbol{s}}\rangle_{12} |R\rangle_R = \frac{1}{\sqrt{\boldsymbol{N}M}} \sum_{|\lambda\rangle\in\Lambda} \sum_{y=0}^{M-1} e^{2\pi i l y\lambda/M} |\lambda\rangle_1 |\lambda\rangle_2 |y\rangle_R$. For details of the implementation of Eq. (20), see Appendix C.

Finally, we perform the quantum Fourier transform $U_{QFT} |y\rangle = \frac{1}{\sqrt{M}} \sum_{p=0}^{M-1} e^{-2\pi i p y/M} |p\rangle$ on the register R, and the state of the system becomes

$$U_{QFT} U_B |\tilde{\boldsymbol{s}}\rangle_{12} |R\rangle_R = \frac{1}{M\sqrt{\boldsymbol{N}}} \sum_{\lambda\in\Lambda} \sum_{y=0}^{M-1} \sum_{p=0}^{M-1} e^{2\pi i (l\lambda - p) y/M} |\lambda\rangle_1 |\lambda\rangle_2 |p\rangle_R . \tag{21}$$

Notice that we can sum over y to rewrite the coefficients in Eq. (21) as $\dfrac{e^{2\pi i l\lambda} - 1}{e^{2\pi i (l\lambda - p)/M} - 1}$. Moreover, these coefficients are strongly peaked at $p \approx l\lambda$, and at the peaks, the coefficients are approximately equal to M. Therefore, $U_{QFT} U_B |\tilde{\boldsymbol{s}}\rangle_{12} |R\rangle_R \approx \frac{1}{\sqrt{\boldsymbol{N}}} \sum_{|\lambda\rangle\in\Lambda} |\lambda\rangle_1 |\lambda\rangle_2 |l\lambda\rangle_R$.

A measurement of the register R yields p with probability $\mathcal{P}(p) = \| {}_R\langle p| U_{QFT} U_B |\tilde{\boldsymbol{s}}\rangle_{12} |R\rangle_R \|^2$ which can be written as

$$\mathcal{P}(p) = \frac{1}{\boldsymbol{N}} \sum_{\lambda} g_\lambda(p) \ , \ \text{with} \quad g_\lambda(p) = \frac{1}{M^2} \frac{\sin^2 \pi l\lambda}{\sin^2 \frac{\pi(l\lambda - p)}{M}} \ . \tag{22}$$

Approximately, the probability $\mathcal{P}(p)$ vanishes for all p, except at $p \approx l\lambda$. At the peaks, each eigenvalue contributes $1/\boldsymbol{N}$. Therefore, each peak is approximately proportional to the multiplicity of the corresponding eigenvalue. In particular, for $p = l\xi$, which corresponds to the eigenvalue of interest $\lambda = \xi$, the probability in Eq. (22) becomes

$$\mathcal{P}(l) = \frac{\beta_k^{\epsilon,\epsilon'}}{\boldsymbol{N}} + \frac{1}{M^2 \boldsymbol{N}} \sum_{\lambda\neq\xi} \frac{\sin^2 \pi l\lambda}{\sin^2 \frac{\pi l(\lambda - 1)}{M}} . \tag{23}$$

In the limit of Eq. (23) as $M \to \infty$, the sum over eigenvalues $\lambda \neq \xi$ vanishes, and we obtain $\beta_k^{\epsilon,\epsilon'} = \boldsymbol{N}\mathcal{P}(l\xi)$.

If one is interested in the whole spectrum of the persistent combinatorial Laplacian, M and l must be chosen in a way that the positive spectrum of the shifted Dirac operator is covered with the different values of p. Notice that each eigenvalue is approximated as $\lambda \approx \frac{p}{l}$. If we are only interested in the harmonic spectra of the persistent combinatorial Laplacian, we only need to capture the peak at $p \approx l\xi$ and make sure it is resolved from other neighboring peaks.

4 An application

This Section demonstrates how our algorithm computes persistence Betti numbers, which track topological features across different scales. This extends previous work in [40] and [41] where the proposed quantum algorithms calculated Betti numbers only without addressing persistence features. To do this, we apply our method to the data set suggested in [48] and described below for the sake of completeness.

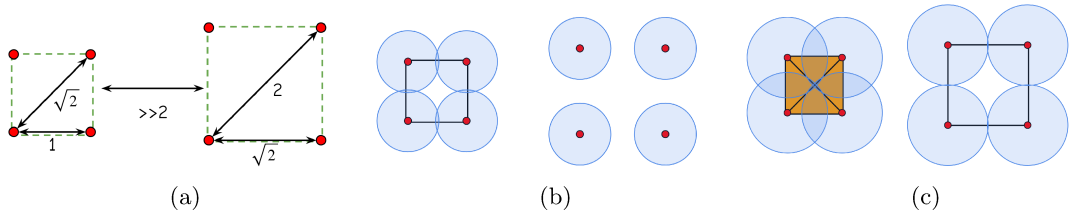

Fig. 3: (a) A two square point cloud data set with the same characteristics as featured in [48]; Construction of the VR at two different scales. Precisely, (b) at ϵ_1; and (c) at $\epsilon_2 > \epsilon_1$.

Consider a point cloud of 8 points consisting of two well-separated squares, as in Figure 3a. The smaller square has sides of length 1, while the larger square has sides of length $\sqrt{2}$. The distance between the two squares exceeds 2. It is easy to see that the smaller square produces a loop in the VR complex at scale 1 (Figure 3b) which disappears at scale $\sqrt{2}$, while at the same time the larger square produces a new loop (Figure 3c). It follows that for $1 < \epsilon_1 < \sqrt{2} < \epsilon_2 < 2$, the one–dimensional persistent Betti numbers corresponding to the point cloud of Figure 3 are

$$\beta_1^{\epsilon_1,\epsilon_1} = 1 \ , \ \beta_1^{\epsilon_2,\epsilon_2} = 1 \ , \ \beta_1^{\epsilon_1,\epsilon_2} = 0 \ . \tag{24}$$

It should be pointed out that the algorithms proposed in [40, 41] can detect the number of loops in the VR complex at scales ϵ_1 and ϵ_2 by computing $\beta_1^{\epsilon_1,\epsilon_1}$ and $\beta_1^{\epsilon_2,\epsilon_2}$, respectively. However these Betti numbers do not hold any persistence information. The algorithms in [40, 41] cannot calculate the persistence Betti number $\beta_1^{\epsilon_1,\epsilon_2}$ which holds the persistence information (number of loops present at scale ϵ_1 that persist to scale ϵ_2). Thus, these algorithms cannot track topological features across different scales. Moreover,

even though $\beta_1^{\epsilon_1,\epsilon_1} = \beta_1^{\epsilon_2,\epsilon_2}$, the persistence Betti number cannot be deduced from $\beta_1^{\epsilon_1,\epsilon_1}$ and $\beta_1^{\epsilon_2,\epsilon_2}$. This is because it indicates that the loops at scales across $\sqrt{2}$ are different, which is additional information to the existence of a loop encoded in the other two Betti numbers.

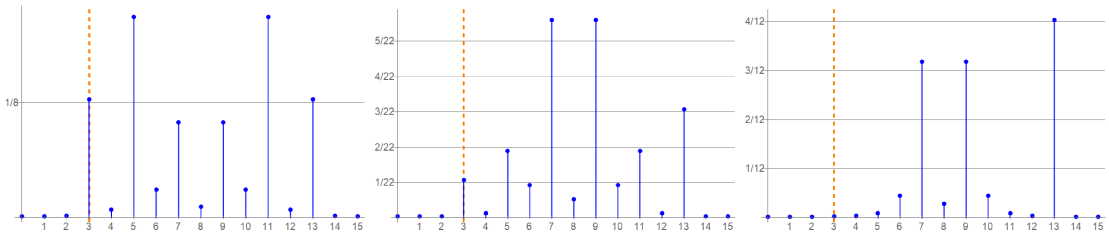

Fig. 4: Probability density corresponding to the persistent Dirac operator (a) $B_1^{\epsilon_1,\epsilon_1}[\xi]$, (b) $B_1^{\epsilon_2,\epsilon_2}[\xi]$, and (c) $B_1^{\epsilon_1,\epsilon_2}[\xi]$, with $\xi = 1$, $l = 3$, $M = 16$. The heights at $p = 3$, multiplied by the dimension of the Hilbert space (8, 22, and 12, respectively), yield the Betti numbers $\beta_1^{\epsilon_1,\epsilon_1} = \beta_1^{\epsilon_2,\epsilon_2} = 1$, and the persistent Betti number $\beta_1^{\epsilon_1,\epsilon_2} = 0$.

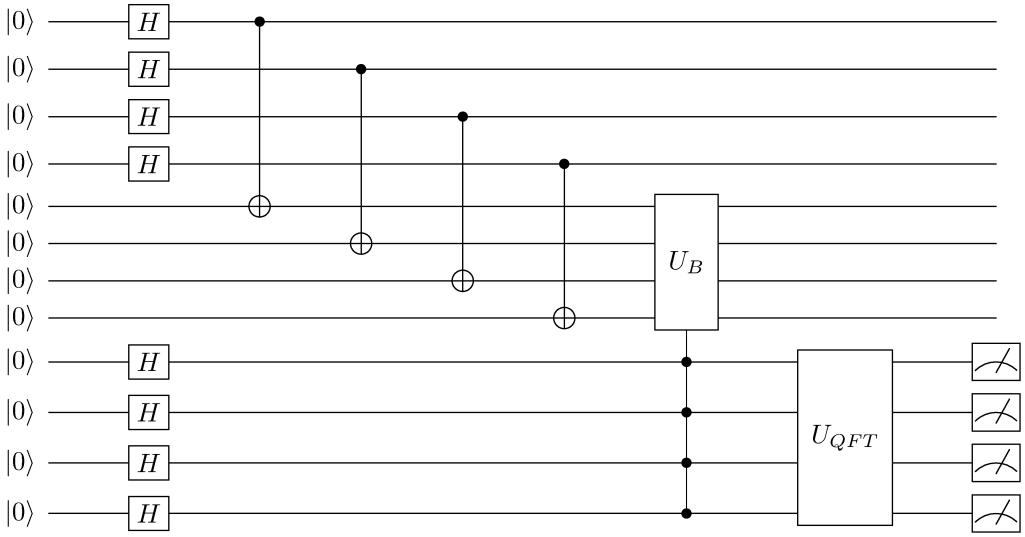

Fig. 5: Quantum circuit for the calculation of Betti number $\beta_1^{\epsilon_1,\epsilon_2}$.

Figure 5 shows the quantum circuit for the calculation of Betti number $\beta_1^{\epsilon_1,\epsilon_2}$. The Dirac operator $B_1^{\epsilon_1,\epsilon_2}[\xi]$ acts on an 12-dimensional Hilbert space. For $\xi = 1$, the eigenvalues are $\lambda = -1, \pm\sqrt{5}$, each of degeneracy 4. Notice that 1 is not an eigenvalue, therefore $\beta_1^{\epsilon_1,\epsilon_2} = 0$. This is confirmed by the probabilities from the quantum algorithm depicted in Figure 5 as explained below.

A register of 4 qubits initially in the state $|0000\rangle_1$ is brought into the state $\frac{1}{4}\sum_{x=0}^{15}|x\rangle_1$, by acting with the Hadamard matrix on each. Then we use them as control to apply CNOT on each of 4 qubits

in an additional register in the state $|0000\rangle_2$, thus entangling them to the state $\frac{1}{4}\sum_{x=0}^{15}|x\rangle_1|x\rangle_2$. We introduce a third register of 4 qubits (choosing $M = 16$) in the state $|0000\rangle_R$ and act on each with the Hadamard gate to bring them into the state $\frac{1}{4}\sum_{y=0}^{15}|y\rangle_R$. We then use them as control to act on register 2 with the exponential of the shifted Dirac operator (see Section 3.4 for details). Finally, we measure all 4 qubits in the register R. The result is the probability distribution $\mathcal{P}(p)$, where $p = 0, 1, \ldots, 15$ (Eq. (22)) depicted in Figure 4c. With the choice $l = 3$, $M = 16$, a measurement of the register of 4 qubits yields no peak at $p = 3$, showing that $\beta_1^{\epsilon_1,\epsilon_2} = 0$.

The persistent Betti number $\beta_1^{\epsilon_1,\epsilon_1}$ is calculated using an eight-dimensional Hilbert space. The eigenvalues of the persistent Dirac operator for $\xi = 1$ are $\pm 1, \pm\sqrt{3}, \pm\sqrt{5}$. Two of the eigenvalues ($\pm\sqrt{3}$) are degenerate with multiplicity 2. We are interested in the multiplicity of the eigenvalue $\lambda = \xi = 1$, which is shown to be 1 by the peak at $p = 3$ of height 1/8 (see Fig.4a). The closest eigenvalue to $\lambda = 1$ is $\lambda = \sqrt{3}$ which is near $p = 5$. Thus, the peak at the nearest eigenvalue is well separated from the one of interest (in Figure 4a), one can see a dip at $p = 4$, and the height $\mathcal{P}(5) \approx \frac{2}{8}$, confirming the double degeneracy of the eigenvalue $\lambda = \sqrt{3}$). The quantum circuit flows as the one in Figure 5 but with registers 1 and 2 having 3 qubits each.

The calculation of Betti number $\beta_1^{\epsilon_2,\epsilon_2}$ proceeds similarly. The Hilbert space is 22-dimensional and for $\xi = 1$, the eigenvalues are the same as for $\beta_1^{\epsilon_1,\epsilon_1}$, but with degeneracies 1, 3, 2, 7 for $\lambda = 1, -1, \pm\sqrt{3}, \pm\sqrt{5}$, respectively. The distance from the eigenvalue of interest ($\lambda = 1$) to its closest one ($\lambda = \sqrt{3}$) is same as before, therefore we can choose $l = 3, M = 16$, again. The quantum circuit is similar to the one in Figure 5 except that registers 1 and 2 need 5 qubits each. The resulting probability distribution is depicted in Figure 4b showing that $\beta_1^{\epsilon_2,\epsilon_2} = 1$.

5 Conclusion

Our work established a quantum persistent homology algorithm that detected and computed the topological features of point cloud data as their resolution changes. Our method considered the persistent Dirac operator, generalizing the Dirac operator discussed in [40, 41]. We provided an implementation on the challenging problem of two particular point cloud squares proposed in [48] as an example in which persistent Betti numbers cannot be deduced from Betti numbers. Our algorithm can encode the full simplicial complex with only $\mathcal{O}(n)$ qubits, using exponentially less memory than a classical counterpart However, it is unlikely that the same can be true about the number of operations [58–60]. In particular, the depth of Grover's search algorithm required to implement projections depends on a ratio of the dimensions of the subspaces $\mathcal{H}^{\epsilon}$, $\mathcal{H}^{\epsilon'}$ and $\mathcal{H}^{\epsilon,\epsilon'}$ to the whole Hilbert space $\mathcal{H}$, which could result in an exponential depth as discussed in [60]. Still, our algorithm could provide at least Grover-like speedup over classical

algorithms. In addition, classical algorithms often have issues computing features of higher dimensions, which is where most quantum algorithms including ours obtain the greatest advantage [60]. Moreover, our algorithm can also recover non-harmonic spectra from the persistent combinatorial Laplacian, while other quantum algorithms for persistent homology [49, 50] can only estimate the dimension of its kernel.

Last, our algorithm is flexible enough and could be extended to consider other type of data, including time series data sets, e.g. see in [61].

Acknowledgments. The authors would like to thank two anonymous reviewers for their comments that substantially imporved this paper. Research supported by the National Science Foundation award DMS-2012609. G. Siopsis also acknowledges the Army Research Office award W911NF-19-1-0397, and the National Science Foundation award OMA-1937008.

Appendix A Membership Oracle

We can encode the order k of a simplex σ in a state $|k\rangle$ $(k = 0, 1, \ldots, n-1)$ by starting from $|0\rangle$ and performing permutations $0 \to 1 \to \cdots \to n-1 \to 0$, conditional upon the corresponding digit of σ being 1. Thus, we perform k permutations mapping $|0\rangle \to |k\rangle$. This can be implemented efficiently because the permutation is a 1-sparse matrix.

To encode the scale ϵ we need information on the data points that can be stored in quantum parallel in QRAM, if it is available, and accessed efficiently [62, 63]. For any $i, j = 1, 2, \ldots, n$, $\text{QRAM}|i\rangle|j\rangle|0\rangle = |i\rangle|j\rangle|d(i,j)\rangle$, where $d(i,j)$ is the distance between points i and j. Notice that the size of the memory is only logarithmic on the number of data points. We introduce a register of qubits to record the parameter ϵ as $|\epsilon\rangle$. We need to know when $d(i,j) \le \epsilon$ to form a VR complex. This information will be stored in a qubit initially in the state $|0\rangle$, and flipped if the membership condition is satisfied. This is implemented with a unitary test that uses the qubit registers storing $d(i,j)$ and ϵ as controls to flip the last qubit,

$$U^{\epsilon}_{\text{test}}|d(i,j)\rangle|\epsilon\rangle|0\rangle = |d(i,j)\rangle|\epsilon\rangle|a^{\epsilon}(i,j)\rangle \ , \quad a^{\epsilon}(i,j) = \begin{cases} 0 \ , \ d(i,j) > \epsilon \\ 1 \ , \ d(i,j) \le \epsilon \end{cases} \tag{A1}$$

Next, in order to know if $\sigma \in S^{\epsilon}$, we must check if $d(i,j) \le \epsilon$ for all (i,j) pairs such that $v_i = v_j = 1$. To this end, we make $\mathcal{O}(k^2)$ calls to QRAM, where k is the dimension of σ. For each pair (i,j), we use $|\sigma\rangle$ as control to call QRAM and apply the test provided $v_i = v_j = 1$, $\text{QRAM}^{\dagger}U^{\epsilon}_{\text{test}}\text{QRAM}|\sigma\rangle|i\rangle|j\rangle|0\rangle|\epsilon\rangle|0\rangle = |\sigma\rangle|i\rangle|j\rangle|0\rangle|\epsilon\rangle|a^{\epsilon}(i,j)\rangle$. The membership of σ in the VR complex, S^{ϵ}, is decided if for all (i,j) we end up with $a^{\epsilon}(i,j) = 1$.

Appendix B Grover's algorithm

Here we review the salient features of amplitude amplification [56] and Grover's search algorithm [57] which are needed for the implementation of the projection P^ϵ (Eq. (12)), for completeness.

Let $|\Psi_k\rangle \in \mathcal{H}_k$ be a state in the span of the k-simplex states. We wish to construct the normalized projected state $|\Psi_k^\epsilon\rangle = \frac{P^\epsilon|\Psi_k\rangle}{\|P^\epsilon|\Psi_k\rangle\|} \in \mathcal{H}_k^\epsilon$, assuming that it exists. To this end, we introduce the unitary operator $U_G = -U_{\Psi_k}U^\epsilon$, with $U_{\Psi_k} = I - 2|\Psi_k\rangle\langle\Psi_k|$ and $U^\epsilon = I - 2P^\epsilon$. Since $\mathcal{H}_k^\epsilon$ is a closed subspace, we may write $|\Psi_k\rangle$ as $|\Psi_k\rangle = \sin\theta\,|\Psi_k^\epsilon\rangle + \cos\theta\,|\bar{\Psi}_k^\epsilon\rangle$, where $|\Psi_k^\epsilon\rangle \in \mathcal{H}_k^\epsilon$ and $|\bar{\Psi}_k^\epsilon\rangle \in \mathcal{H}_k^{\epsilon\perp}$. Notice that $\sin\theta = \|P^\epsilon|\Psi_k\rangle\|$. We can think of $|\Psi_k\rangle$ as the vector $(\sin\theta, \cos\theta)^T$ in the two-dimensional space spanned by $\{|\Psi_k^\epsilon\rangle, |\bar{\Psi}_k^\epsilon\rangle\}$, then U_G acts as a rotation by an angle 2θ. Applying it K times, we obtain the state $U_G^K|\Psi_k\rangle = \sin(2K+1)\theta\,|\Psi_k^\epsilon\rangle + \cos(2K+1)\theta\,|\bar{\Psi}_k^\epsilon\rangle$. This is close to the desired state for $(2K+1)\theta \approx \frac{\pi}{2}$. Therefore, the number of Grover steps needed is $K = \lfloor\frac{\pi}{4\theta}\rfloor$. As discussed in [60], K could be exponential on the number of data points if $\|P^\epsilon|\Psi_k\rangle\|$ is small, for example when the number of simplices present in S_k^ϵ is only polynomial on the number data points.

Appendix C Implementation of an exponential operator

Relying on [64], we review the construction of the exponential operator $e^{itB_k^{\epsilon,\epsilon'}[\xi]}$, where the shifted Dirac matrix $B_k^{\epsilon,\epsilon'}[\xi]$ is defined in Eq. (17). This construction is needed for the phase estimation algorithm described in Section 3.4 (See Eq. (20)). We start by constructing the SWAP_B operator from the shifted Dirac operator $B_k^{\epsilon,\epsilon'}[\xi]$,

$$\mathcal{S} \equiv \text{SWAP}_B = \sum_{\psi,\phi} B_k^{\epsilon,\epsilon'}[\xi](\psi,\phi)\,|\phi\rangle\langle\psi| \otimes |\psi\rangle\langle\phi| \quad , \quad B_k^{\epsilon,\epsilon'}[\xi](\psi,\phi) = \langle\psi|\,B_k^{\epsilon,\epsilon'}[\xi]\,|\phi\rangle \quad , \tag{C2}$$

where $|\psi\rangle = |0\rangle|\psi_0\rangle + |1\rangle|\psi_1\rangle + |2\rangle|\psi_2\rangle$, with $|\psi_0\rangle \in \mathcal{H}_{k-1}$, $|\psi_1\rangle \in \mathcal{H}_k$, $|\psi_2\rangle \in \mathcal{H}_{k+1}$, and similarly for $|\phi\rangle$. Let $\boldsymbol{N}$ be the dimensionality of the Hilbert space in which $|\psi\rangle$ and $|\phi\rangle$ live and $\{|e_a\rangle, a = 1, \ldots, \boldsymbol{N}\}$ an orthonormal basis for the Hilbert space under consideration. With the choice $\xi = 1$, all matrix elements of the $\boldsymbol{N} \times \boldsymbol{N}$ matrix $B_k^{\epsilon,\epsilon'}[\xi](e_a, e_b) \in \{0, \pm 1\}$ and the matrix $\mathcal{S}$ can be efficiently constructed. Then we construct the exponential SWAP_B operator $e^{i\Delta t\mathcal{S}}$, which can be done efficiently because $\mathcal{S}$ is a one-sparse matrix. Next, we act on the state $|\boldsymbol{s}\rangle \otimes |\Psi\rangle$, where $|\boldsymbol{s}\rangle$ is the uniform state

$$|\boldsymbol{s}\rangle = \frac{1}{\sqrt{\boldsymbol{N}}}\sum_{a=1}^{\boldsymbol{N}} |e_a\rangle\,, \tag{C3}$$

and $|\Psi\rangle$ is an arbitrary state in the subspace on which $B_k^{\epsilon,\epsilon'}[\xi]$ acts. After tracing over the space in which $|\boldsymbol{s}\rangle$ lives, we obtain $\mathrm{tr}_1\left[e^{-i\Delta t\mathcal{S}}\,|\boldsymbol{s}\rangle\,\langle\boldsymbol{s}|\otimes|\Psi\rangle\,\langle\Psi|\,e^{i\Delta t\mathcal{S}}\right] = e^{-iB_k^{\epsilon,\epsilon'}[\xi]\Delta t/\boldsymbol{N}}\,|\Psi\rangle\,\langle\Psi|\,e^{iB_k^{\epsilon,\epsilon'}[\xi]\Delta t/\boldsymbol{N}} + \mathcal{O}(\Delta t^2)$, which projects onto the state $e^{-iB_k^{\epsilon,\epsilon'}[\xi]\Delta t/\boldsymbol{N}}\,|\Psi\rangle$ up to second order in Δt. The desired state $e^{-itB_k^{\epsilon,\epsilon'}[\xi]}\,|\Psi\rangle$ for finite t can be obtained by repeating the above construction as many times as needed.

References

[1] G. Carlsson, Topology and data. Bulletin of the American Mathematical Society **46**(2), 255–308 (2009)

[2] E. Carlsson, G. Carlsson, V.D. Silva, An algebraic topological method for feature identification. International Journal of Computational Geometry **16**(4), 291–314 (2006)

[3] H. Edelsbrunner, J. Harer, Computational topology: an introduction (American Mathematical Society, 2010)

[4] R. Vasudevan, A. Ames, R. Bajcsy, Persistent homology for automatic determination of human-data based cost of bipedal walking. Nonlinear Analysis: Hybrid Systems **7**(1), 101–115 (2013)

[5] G. Kusano, K. Fukumizu, Y. Hiraoka, Persistence weighted Gaussian kernel for topological data analysis. arXiv:1601.01741 (2016)

[6] H. Edelsbrunner, in International Workshop on Graph-Based Representations in Pattern Recognition (Springer, 2013), pp. 182–183

[7] M. Chung, J. Hanson, J. Ye, R. Davidson, S. Pollak, Persistent homology in sparse regression and its application to brain morphometry. IEEE transactions on medical imaging **34**(9), 1928–1939 (2015)

[8] E. Munch, Applications of persistent homology to time varying systems. Ph.D. thesis, Duke University (2013)

[9] F.A. Khasawneh, E. Munch, Chatter detection in turning using persistent homology. Mechanical Systems and Signal Processing **70**, 527–541 (2016)

[10] V. Venkataraman, K.N. Ramamurthy, P. Turaga, Persistent homology of attractors for action recognition. arXiv:1603.05310 (2016)

[11] A. Adcock, E. Carlsson, G. Carlsson, The ring of algebraic functions on persistence bar codes. Homology, Homotopy and Applications **18**(1), 381–402 (2016)

[12] S. Emrani, T. Gentimis, H. Krim, Persistent homology of delay embeddings and its application to wheeze detection. IEEE Signal Processing Letters **21**(4), 459–463 (2014)

[13] C.M. Pereira, R.F. de Mello, Persistent homology for time series and spatial data clustering. Expert Systems with Applications **42**(15), 6026–6038 (2015)

[14] M. Nicolau, A.J. Levine, G. Carlsson, Topology based data analysis identifies a subgroup of breast cancers with a unique mutational profile and excellent survival. Proceedings of the National Academy of Sciences **108**(17), 7265–7270 (2011)

[15] J. Mike, V. Maroulas, Combinatorial hodge theory for equitable kidney paired donation. Foundations of Data Science **1**(1), 87–101 (2019)

[16] J. Mike, C.D. Sumrall, V. Maroulas, F. Schwartz, Nonlandmark classification in paleobiology: computational geometry as a tool for species discrimination. Paleobiology pp. 1–11 (2016)

[17] V. Maroulas, F. Nasrin, C. Oballe, A bayesian framework for persistent homology. SIAM Journal on Mathematics of Data Science **2**(1), 48–74 (2020)

[18] J. Townsend, C. Micucci, J. Hymel, V. Maroulas, K.D. Vogiatzis, Representation of molecular structures with persistent homology for machine learning applications in chemistry. Nat Commun **11**, 3230 (2020)

[19] V. Maroulas, C.P. Micucci, F. Nasrin, Bayesian Topological Learning for Classifying the Structure of Biological Networks. Bayesian Analysis pp. 1 – 26 (2021)

[20] J.A. Perea, J. Harer, Sliding windows and persistence: An application of topological methods to signal analysis. Foundations of Computational Mathematics **15**(3), 799–838 (2015)

[21] D. Rouse, A. Watkins, D. Porter, J. Harer, P. Bendich, N. Strawn, E. Munch, J. DeSena, J. Clarke, J. Gilbert, in SPIE Defense+ Security (International Society for Optics and Photonics, 2015), pp. 94740L–94740L

[22] V. De Silva, R. Ghrist, Coordinate-free coverage in sensor networks with controlled boundaries via homology. The International Journal of Robotics Research **25**(12), 1205–1222 (2006)

[23] V. De Silva, R. Ghrist, Homological sensor networks. Notices of the American mathematical society **54**(1) (2007)

[24] P. Dlotko, R. Ghrist, M. Juda, M. Mrozek, Distributed computation of coverage in sensor networks by homological methods. Applicable Algebra in Engineering, Communication and Computing **23**(1-2), 29–58 (2012)

[25] G. Carlsson, V. De Silva, Zigzag persistence. Foundations of computational mathematics **10**(4), 367–405 (2010)

[26] G. Carlsson, V. De Silva, D. Morozov, in Proceedings of the twenty-fifth annual symposium on Computational geometry (ACM, 2009), pp. 247–256

[27] V. De Silva, R. Ghrist, Coverage in sensor networks via persistent homology. Algebraic & Geometric Topology **7**(1), 339–358 (2007)

[28] J. Binchi, E. Merelli, M. Rucco, G. Petri, F. Vaccarino, jholes: A tool for understanding biological complex networks via clique weight rank persistent homology. Electronic Notes in Theoretical Computer Science **306**, 5–18 (2014)

[29] K. Xia, X. Feng, Y. Tong, G.W. Wei, Persistent homology for the quantitative prediction of fullerene stability. Journal of computational chemistry **36**(6), 408–422 (2015)

[30] M. Guillemard, A. Iske, Signal filtering and persistent homology: an illustrative example. Proc. Sampling Theory and Applications (SampTA'11) (2011)

[31] P. Bendich, J.S. Marron, E. Miller, A. Pieloch, S. Skwerer, Persistent homology analysis of brain artery trees. The Annals of Applied Statistics **10**(1), 198–218 (2016)

[32] A. Marchese, V. Maroulas, in Information Fusion (FUSION), 2016 19th International Conference on (ISIF, 2016), pp. 1377–1381

[33] A. Marchese, V. Maroulas, Signal classification with a point process distance on the space of persistence diagrams. Advances in Data Analysis and Classification **12**(3), 657–682 (2018)

[34] L.M. Seversky, S. Davis, M. Berger, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops (2016), pp. 59–67

[35] J.A. Perea, A. Deckard, S.B. Haase, J. Harer, Sw1pers: Sliding windows and 1-persistence scoring; discovering periodicity in gene expression time series data. BMC bioinformatics **16**(1), 257 (2015)

[36] D. Morozov. Dionysus, a C++ library for computing persistent homology (2007)

[37] U. Bauer. Ripser (2015). Https://github.com/Ripser/ripser

[38] C. Chen, M. Kerber, in Proceedings 27th European Workshop on Computational Geometry, vol. 11 (2011)

[39] A. Zomorodian, G. Carlsson, Computing persistent homology. Discrete & Computational Geometry **33**(2), 249–274 (2005)

[40] S. Lloyd, S. Garnerone, P. Zanardi, Quantum algorithms for topological and geometric analysis of data. Nature communications **7**(1), 1–7 (2016)

[41] G. Siopsis, Quantum topological data analysis with continuous variables. Foundations of Data Science **4**(1), 419–431 (2019)

[42] H. Huang, *et al.*, Demonstration of topological data analysis on a quantum processor. Optica **5**, 193 (2018)

[43] R. Dridi, H. Alghassi, Homology computation of large point clouds using quantum annealing. arXiv:1512.09328 (2016)

[44] C.R. Wie, A quantum circuit to construct all maximal cliques using grover search algorithm. arXiv preprint arXiv:1711.06146 (2017)

[45] S. Ubaru, I.Y. Akhalwaya, M.S. Squillante, K.L. Clarkson, L. Horesh. Quantum topological data analysis with linear depth and exponential speedup (2021)

[46] I.Y. Akhalwaya, S. Ubaru, K.L. Clarkson, M.S. Squillante, V. Jejjala, Y.H. He, K. Naidoo, V. Kalantzis, L. Horesh. Towards quantum advantage on noisy quantum computers (2022)

[47] D.W. Berry, Y. Su, C. Gyurik, R. King, J. Basso, A.D.T. Barba, A. Rajput, N. Wiebe, V. Dunjko, R. Babbush. Analyzing prospects for quantum advantage in topological data analysis (2023)

[48] S. Gunn, N. Kornerup, Review of a quantum algorithm for betti numbers. arXiv preprint arXiv:1906.07673 (2019)

[49] R. Hayakawa, Quantum algorithm for persistent betti numbers and topological data analysis. arXiv preprint arXiv:2111.00433 (2021)

[50] S. McArdle, A. Gilyén, M. Berta. A streamlined quantum algorithm for topological data analysis with exponentially fewer qubits (2022)

[51] R. Wang, D.D. Nguyen, G.W. Wei, Persistent spectral graph. International journal for numerical methods in biomedical engineering **36**(9), e3376 (2020)

[52] F. Mémoli, Z. Wan, Y. Wang, Persistent laplacians: Properties, algorithms and implications. SIAM Journal on Mathematics of Data Science **4**(2), 858–884 (2022)

[53] Z. Meng, K. Xia, Persistent spectral–based machine learning (perspect ml) for protein-ligand binding affinity prediction. Science Advances **7**(19), eabc5329 (2021). https://doi.org/10.1126/sciadv.abc5329. URL https://www.science.org/doi/abs/10.1126/sciadv.abc5329. https://www.science.org/doi/pdf/10.1126/sciadv.abc5329

[54] J. Chen, Y. Qiu, R. Wang, G.W. Wei, Persistent laplacian projected omicron ba. 4 and ba. 5 to become new dominating variants. Computers in Biology and Medicine **151**, 106262 (2022)

[55] I.Y. Akhalwaya, Y.H. He, L. Horesh, V. Jejjala, W. Kirby, K. Naidoo, S. Ubaru, Representation of the fermionic boundary operator. Phys. Rev. A **106**, 022407 (2022). https://doi.org/10.1103/PhysRevA.106.022407. URL https://link.aps.org/doi/10.1103/PhysRevA.106.022407

[56] G. Brassard, P. Hoyer, M. Mosca, A. Tapp, Quantum amplitude amplification and estimation. Contemporary Mathematics **305**, 53–74 (2002)

[57] L.K. Grover, Quantum computers can search rapidly by using almost any transformation. Phys. Rev. Lett. **80**, 4329–4332 (1998). https://doi.org/10.1103/PhysRevLett.80.4329. URL https://link.aps.org/doi/10.1103/PhysRevLett.80.4329

[58] M. Crichigno, T. Kohler, Clique homology is qma1-hard. arXiv preprint arXiv:2209.11793 (2022)

[59] C. Cade, P.M. Crichigno, Complexity of supersymmetric systems and the cohomology problem. arXiv preprint arXiv:2107.00011 (2021)

[60] A. Schmidhuber, S. Lloyd, Complexity-theoretic limitations on quantum algorithms for topological data analysis. arXiv preprint arXiv:2209.14286 (2022)

[61] B. Ameneyro, G. Siopsis, V. Maroulas, in 2022 IEEE/ACM 7th Symposium on Edge Computing (SEC) (IEEE, 2022), pp. 387–392

[62] V. Giovannetti, S. Lloyd, L. Maccone, Quantum random access memory. Phys. Rev. Lett. **100**, 160501 (2008). https://doi.org/10.1103/PhysRevLett.100.160501. URL https://link.aps.org/doi/10.1103/PhysRevLett.100.160501

[63] V. Giovannetti, S. Lloyd, L. Maccone, Architectures for a quantum random access memory. Physical Review A **78**(5), 052310 (2008)

[64] P. Rebentrost, A. Steffens, I. Marvian, S. Lloyd, Quantum singular-value decomposition of nonsparse low-rank matrices. Physical review A **97**(1), 012327 (2018)