Distributed Coverage Verification in Sensor Networks Without Location Information

Alireza Tahbaz-Salehi and Ali Jadbabaie

Abstract

In this paper, we present a series of distributed algorithms for coverage verification in sensor networks with no location information. We demonstrate how, in the absence of localization devices, simplicial complexes and tools from computational homology can be used in providing valuable information on the properties of the cover. Our approach is based on computation of certain generators of the first homology of the Rips complex corresponding to the sensor network. We use the theory of higher order combinatorial Laplacian operators to compute generators of the homology classes of the Rips complex in a distributed fashion. In particular, we first present a decentralized scheme based on Laplacian flows to compute a generator of the first homology, which represents coverage holes. Then we formulate the problem of localizing coverage holes as an optimization problem to compute the sparsest generator of the first homology classes of the Rips complex. Furthermore, we show that one can detect redundancies in the sensor network by finding the sparsest generator of the second homology of the cover relative to its boundary. We also demonstrate how subgradient methods can be used in solving these optimization problems in a distributed manner. Finally, non-trivial simulations are provided that illustrate the performance of our algorithms.

I. INTRODUCTION

Recent advances in computing, communication, sensing and actuation technologies, have brought networks composed of hundreds or even thousands of inexpensive mobile sensing platforms closer to reality. This has induced a significant amount of interest in development of analytical tools for predicting the behavior, as well as controlling the complexities of such large-scale sensor networks. Designing algorithms for deployment, localization, duty-cycling,
communication and coverage verification in sensor networks form the core of this active area of research.

Of the most fundamental problems in this domain is the coverage problem. In general, this reflects how well an area of interest is monitored or tracked by sensors. In most applications, we are interested in a reliable coverage of the environment in such a way that there are no gaps left in the coverage. Algorithms for this purpose have been extensively studied [1]. One of the most prominent approaches for addressing the coverage problem has been the ‘computational geometry’ approach, in which one uses the coordinates of the nodes and standard geometric tools (such as Delaunay triangulations or Voronoi diagrams) to determine coverage [2]–[5]. One very well-known example of utilizing this geometric approach is in solving the Art Gallery Problem, in which one determines the number of observers necessary to cover an art gallery (or an area of interest) such that every point in the gallery is monitored by at least one observer [6], [7].

Such geometrical approaches often suffer from the drawback that they can be too expensive to compute in real-time. Moreover, in most applications, they require exact knowledge of the locations of the sensors. Although, this information can be made available in real-time by a localization algorithm or by the means of localization devices (such as GPS), it can only be used most effectively in an off-line pre-deployment analysis for large networks or when there are strong assumptions on geometrical structure of the network and the environment. This drawback becomes more evident if the network topology changes due to node mobility or sensor failure. In such cases, a continuous monitoring of the network coverage becomes prohibitive if the algorithm is too expensive to run or is sensitive to location uncertainty. Finally, localization equipments add to the cost of the network, which can be a limiting factor as the size of the network grows. Consequently, a minimal geometry approach for addressing these issues becomes essential.

More recently, topological spaces and their topological invariants have been used in addressing the coverage problem in the absence of geometric data, such as location or orientation [8]–[14]. One notable characteristic of these studies is the use of topological abstractions which preserve many global geometrical properties of the network while abstracting away the small scale redundant details. For instance, in [8], the authors construct the Rips complex corresponding to the communication graph of the network and use the fact that the first homology of this simplicial complex provides sufficient information about coverage. This is followed by [9] and [11], in which a relative homological criterion for coverage is presented. These results are further
extended in [10] to networks without boundary, the pursuit-evasion problem and *barrier coverage*. The first steps for implementation of the above mentioned ideas as a distributed algorithm are taken in [12] and [13]. The authors show that the combinatorial Laplacians are the right tools for distributed computation of the elements of the homology groups, and hence, can be used for decentralized coverage verification. They present a consensus-like scheme based on a dynamical system whose stability properties determine the existence of coverage holes, although it fails to locate them. This idea is further extended to time-varying networks for verification of *sweep coverage* in [14].

The contribution of this paper is twofold. First, based on the ideas in [10] and [12], we present a distributed algorithm which is capable of “localizing” coverage holes in a network of sensors without any metric information. More precisely, following [10], [12], we use tools from algebraic topology to represent the coverage properties of the sensor network by its Rips complex. We show that given a generator in the first homology of the Rips complex, the problem of finding the “tightest” cycle encircling the hole represented by that homology class can be formulated as an integer programming problem. Moreover, we present conditions under which the linear programming relaxation of this integer programming problem is exact and therefore, its solution provides the location of the coverage holes in the simplicial complex *without use of any coordinate information*. This optimization-based approach is a direct generalization of network flow algorithms on graphs to simplicial complexes. Finally, we show that if subgradient methods [15]–[17] are used for solving this relaxation, the updates are distributed in nature and therefore, the computation of the tightest cycle around the holes can be implemented in a distributed fashion.

Our approach is quite interdisciplinary in nature and combines results from multiagent systems, agreement and consensus problems [18], [19], with recent advances in coverage maintenance in sensor networks using computational algebraic topology methods and optimization techniques. Moreover, this novel approach is different from the algorithms presented in [20], [21], where it is explicitly assumed that the simplicial complex is embedded on an orientable surface. It is also more general than the results in [22]: our hole detection algorithm is not limited to Rips complexes, is distributed in nature, and does not use node coordinates.

A second contribution of the paper concerns detecting redundancies in the sensor network. Using tools from algebraic topology, we introduce a novel approach for computing a minimal set of sensors required to cover the entire domain. We formulate the problem of computing the
sparsest generator of the second homology of the Rips complex with respect to its boundary as an integer-programming problem and solve its LP relaxation in a distributed way, using subgradient methods. To the best of our knowledge, such an algorithm has not been proposed in any other study.

The paper is organized as follows. We present the basic setup and assumptions of our model in section II. Sections III is meant to provide a brief review on the concepts of simplicial complexes, their homological properties and combinatorial Laplacian operators. Section IV summarizes the results already known in regards of distributed coverage verification in networks with no metric information. Our main results are presented in section V, in which we show that how, in a distributed fashion, one can “localize” coverage holes in a location-free sensor network by solving a linear programming problem using subgradient methods. We extend this idea to a higher dimension in section VI in order to find a sparse cover of the region. Simulations of the two algorithms are presented in section VII. Finally, section VIII contains our conclusions.

II. PROBLEM FORMULATION

Consider a collection of \( n \) stationary sensors, denoted by \( V \), deployed over a region of interest \( D \subset \mathbb{R}^2 \). We assume that these sensors are equipped with local communication and sensing capabilities: each sensor is only capable of communicating with a limited number of other sensors in its proximity, and has a limited sensing range. Furthermore, we assume a complete absence of localization capabilities and metric information, in the sense that the sensors in this network can determine neither distance nor direction. Under these assumption, we are interested in distributed algorithms for coverage verification. In particular, we are interested in verifying the existence of coverage holes, compute their locations, and detect redundancies in the network.

We adopt the following two frameworks as the coverage models, for which we present our coverage verification algorithms:

1) Simplicial Coverage: In this framework, we assume that each sensor is capable of communicating with other sensors within a radially symmetric domain of radius \( r_b \), called the broadcast disk. As for the coverage, we assume a “capture” modality in which any subset of nodes which are in pairwise communication cover their entire convex hull. In other words, the region covered by the sensors is given by

\[
\mathcal{A}(V) = \bigcup \{\text{conv}(Q)|Q \subseteq V, \max_{v_i, v_j \in Q} \|v_i - v_j\|_2 \leq r_b\}
\]
where $V$ is the set of sensor locations and $v_i$ represents the location of the $i$-th sensor. This model, which is inspired by the results in [23], guarantees that the coverage and communication capabilities of the sensors is limited and is based on proximity.

2) **Symmetric Coverage:** Similar to the previous framework, we assume that each sensor is capable of communicating with other sensors within a distance $r_b$. However, unlike the simplicial coverage model, we assume that each sensor is capable of covering a radially symmetric area of radius $r_c$, known as the *coverage radius*. We refer to the disk covered by a sensor located at point $v \in V$ as the its *coverage disk*. The total region covered by the sensors is given by $U(V) = \bigcup_{v \in V} U_v$, where $U_v = \{ x \in \mathbb{R}^2 : \|x - v\| \leq r_c \}$ is the coverage disk corresponding to the sensor located at point $v$. Clearly, the whole region of interest is covered if $D$ is a subset of $U(V)$. For technical reasons that will become clear in the following sections [8], we assume that $r_b \leq r_c \sqrt{3}$.

In the rest of the paper, we develop the required tools and present algorithms that can verify different coverage properties for the above mentioned frameworks. Before doing so, we need to impose some additional restrictions on the geometry of the domain $D$. We assume that $D$ is connected and compact and its boundary $\partial D$ is connected and piecewise linear. Moreover, to avoid boundary effects, we assume that there are sensors, known as *fence nodes*, located on $\partial D$ such that each such sensor is capable of communicating with its two closest neighbors on $\partial D$ on either side.

### III. Simplicial Complexes, Homology and Combinatorial Laplacians

This section is dedicated to the definition of simplicial complexes and their homological properties as they are the main mathematical tools used in this paper. A thorough treatment of the subject can be found in [24] and [25].

Given a set of points $V$, a *$k$-simplex* (or a simplex of dimension $k$) is an unordered set $\{v_0, v_1, \cdots, v_k\} \subseteq V$ where $v_i \neq v_j$ for all $i \neq j$. A *face* of the $k$-simplex $\{v_0, v_1, \cdots, v_k\}$ is a $(k-1)$-simplex of the form $\{v_0, \cdots, v_{i-1}, v_{i+1}, \cdots, v_k\}$ for some $0 \leq i \leq k$. Clearly, any $k$-simplex has exactly $k+1$ faces.

**Definition 1:** A simplicial complex $X$ is a finite collection of simplices which is closed with respect to inclusion of faces, i.e., if $\sigma \in X$, then all faces of $\sigma$ are also in $X$. 
Fig. 1. A simplicial complex, consisting of 11 vertices (0-simplices), 14 edges (1-simplices), 5 2-simplices and one 3-simplex.

Roughly speaking, a simplicial complex is a generalization of a graph, in the sense that in addition to binary relations between the elements of $V$, it models higher relations between them as well. Note that due to the requirement of closure with respect to the inclusion of the faces, a simplicial complex is different from a hyper graph, in which any subset of the powerset of $V$ can be considered as a hyper edge.

The dimension of a simplicial complex is the maximum dimension of any of its simplices. A subcomplex of $X$ is a simplicial complex $Y \subseteq X$. A particular subcomplex of $X$ is its $k$-skeleton consisting of all simplices of dimension $k$ or less $X^{(k)} = \{ \sigma \in X : \dim \sigma \leq k \}$. Therefore, the 1-skeleton of any non-empty simplicial complex is a graph. Given a graph $G$, its flag complex $F(G)$ is the largest simplicial complex whose 1-skeleton is $G$; every $(k+1)$-clique in $G$ defines a $k$-simplex in $F(G)$.

Given a simplicial complex $X$, two $k$-simplices $\sigma_i$ and $\sigma_j$ are upper adjacent (denoted by $\sigma_i \triangleright \sigma_j$) if both are faces of a $(k+1)$-simplex in $X$. The two $k$-simplices are said to be lower adjacent (denoted by $\sigma_i \triangleleft \sigma_j$) if both have a common face. Having defined the concept of adjacency, one can define the upper and lower adjacency matrices, $A_u^{(k)}$ and $A_l^{(k)}$ respectively, in order to book keep the adjacency relations between the $k$-simplices. The first upper adjacency matrix of a simplicial complex $A_u^{(0)}$ coincides with the well-known notion of the adjacency matrix of the graph capturing its 1-skeleton.

A. Boundary Homomorphism

Let $X$ denote a simplicial complex. Similar to graphs, an orientation can be defined for $X$ by defining an ordering on all of its $k$-simplices. We denote the $k$-simplex $\{v_0, \cdots, v_k\}$ with an ordering by $[v_0, \cdots, v_k]$. For each $k \geq 0$, define $C_k(X)$ to be the vector space whose basis is the set of oriented $k$-simplices of $X$, where a change in the orientation corresponds to a change
in the sign of the coefficient as \([v_0, \cdots, v_i, \cdots, v_j, \cdots, v_k] = -[v_0, \cdots, v_j, \cdots, v_i, \cdots, \cdots, v_k]\). We let \(C_k(X) = 0\), if \(k\) is larger than the dimension of \(X\). Therefore, by definition, elements of \(C_k(X)\), called \(k\)-chains, can be written as finite formal sums \(\sum_j \alpha_j \sigma_j^{(k)}\) where the coefficients \(\alpha_j \in \mathbb{R}\) and \(\sigma_j^{(k)}\) are the oriented \(k\)-simplices of \(X\).\(^1\) Note that \(C_k\) is a finite-dimensional vector space with the number of \(k\)-simplices as its dimension. We now define the boundary map.

**Definition 2:** For an oriented simplicial complex \(X\), the \(k\)-th simplicial boundary map is a homomorphism \(\partial_k : C_k(X) \to C_{k-1}(X)\), which acts on the basis elements of its domain via

\[
\partial_k [v_0, \cdots, v_k] = \sum_{j=0}^{k} (-1)^j [v_0, \cdots, v_{j-1}, v_{j+1}, \cdots, v_k].
\]

(1)

Intuitively, the above operator maps a \(k\)-chain to its faces.

Since for any finite simplicial complex \(C_k(X)\) is a finite dimensional vector space for all \(k\), \(\partial_k\) has a matrix representation. We denote the matrix representation of the \(k\)-th boundary map relative to the bases of \(C_k\) and \(C_{k-1}\) by \(B_k \in \mathbb{R}^{n_k \times n_{k-1}}\), where \(n_k\) is the number of \(k\)-simplices of \(X\). In particular, the matrix representation of the first boundary map \(\partial_1\) is nothing but the 

edge-vertex incidence matrix of a graph which maps edges (1-simplices) to vertices (0-simplices). Finally, using (1), it is an easy exercise to show that

**Lemma 1:** The map \(\partial_k \circ \partial_{k+1} : C_{k+1}(X) \to C_{k-1}(X)\) is uniformly zero for all \(k \geq 1\).

In other words, the boundary of any \(k\)-chain has no boundary.

**B. Simplicial Homology**

Let \(X\) denote a simplicial complex. Consider the following two subspaces of \(C_k(X)\):

\(k\)-cycles : \(\ker \partial_k = \{x \in C_k(X) : \partial_k x = 0\}\)

\(k\)-boundaries : \(\mathrm{im} \partial_{k+1} = \{x \in C_k(X) : \exists y \mathrm{s.t.} x = \partial_{k+1} y\}\)

An element in \(\ker \partial_k\) is a subcomplex without a boundary and therefore represents a \(k\)-dimensional cycle, while the elements in \(\mathrm{im} \partial_{k+1}\) are boundaries of higher dimensional chains, and therefore are known as \(k\)-boundaries. The \(k\)-cycles are the basic objects that count the presence of "\(k\)-dimensional holes" in the simplicial complex. But, certainly, many of the \(k\)-cycles in \(X\) are

\(^1\)To be more precise, this is the definition of \(k\)-chains with coefficients in \(\mathbb{R}\). In most algebraic topology texts such as [24], the \(k\)-chains are defined over integers rather than reals. In such a case, \(C_k(X)\) is defined as a free abelian group with the set of oriented \(k\)-simplices as its basis. However, in this paper, we find it more convenient to define the chains over \(\mathbb{R}\), as in [26].
measuring the same hole; still other cycles do not really detect a hole at all—they bound a subcomplex of dimension \( k + 1 \) in \( X \). In fact, we say two \( k \)-cycles \( \xi \) and \( \eta \) are homologous if their difference is a boundary: \( \xi - \eta \in \text{im} \partial_{k+1} \). Therefore, as far as measuring holes is concerned, homologous cycles are equivalent. Consequently, it makes sense to define the quotient vector space

\[
H_k(X) = \ker \partial_k / \text{im} \partial_{k+1},
\]

known as the \( k \)-th homology of \( X \), as the proper vector space for distinguishing homologous cycles. Note that according to Lemma 1, we have \( \partial_k \circ \partial_{k+1} = 0 \), implying that \( \text{im} \partial_{k+1} \) is a subspace of \( \ker \partial_k \), and therefore, making \( H_k(X) \) a well-defined vector space.\(^2\)

Roughly speaking, when constructing the homology, we are removing the cycles that are boundaries of a higher order subcomplex from the set of all \( k \)-cycles, so that the remaining ones carry information about the \( k \)-dimensional holes of the complex. A more precise way of interpreting (2) is that any element of \( H_k(X) \) is an equivalence class of homologous \( k \)-cycles. Moreover, it inherits the structure of a vector space in the natural way: \([\xi] + [\eta] = [\xi + \eta]\) and \( c[\xi] = [c\xi] \) for \( c \in \mathbb{R} \), where \([\xi]\) represents the equivalence class of all \( k \)-cycles homologous to \( \xi \). Therefore, each non-trivial homology class\(^3\) in a certain dimension identifies a corresponding “hole” in that dimension. In fact, the dimension of the \( k \)-th homology of \( X \) (known as its \( k \)-th Betti number) identifies the number of \( k \)-dimensional holes in \( X \). For example, the dimension of \( H_0(X) \) is the number of connected components of \( X \), while the dimension of \( H_1(X) \) is equal to the number of holes in its 2-skeleton.

**C. Relative Homology**

In some applications, one may need to compute the holes modulo some region of space, such as the boundary. The concept of relative homology is defined for this purpose.

Given a simplicial complex \( X \) and a subcomplex \( A \subset X \), let \( C_k(X, A) \) to be the quotient vector space \( C_k(X) / C_k(A) \). In other words, the chains in \( A \) are trivial in \( C_k(X, A) \). Since the

\(^2\)If we define the \( k \)-chains over integers, then \( \text{im} \partial_{k+1} \) becomes a normal subgroup of \( \ker \partial_k \). In that case, the homology is defined as the quotient group \( H_k = \ker \partial_k / \text{im} \partial_{k+1} \). In the rest of the paper, we will use the term homology group, even when we are in fact referring to a vector space.

\(^3\)By the trivial homology class, we mean the equivalence class of all null-homologous \( k \)-cycles on the simplicial complex.
boundary map \( \partial_k : C_k(X) \to C_{k-1}(X) \) takes \( C_k(A) \) to \( C_{k-1}(A) \), it induces a quotient boundary map \( \bar{\partial}_k : C_k(X, A) \to C_{k-1}(X, A) \). One can verify that the subspaces defined by the kernel and image of the quotient map are well-defined and satisfy \( \text{im} \bar{\partial}_{k+1} \subset \ker \bar{\partial}_k \subset C_k(X, A) \). Therefore, similar to before, one can define the \( k \)-th relative homology as the quotient vector space \( H_k(X, A) = \ker \bar{\partial}_k / \text{im} \bar{\partial}_{k+1} \).

Given the above definition, one can interpret elements of \( H_k(X, A) \) as representatives for relative cycles: \( k \)-chains \( \xi \in C_k(X) \) such that \( \partial_k \xi \in C_{k-1}(A) \). Moreover, such a relative cycle \( \xi \) is trivial in \( H_k(X, A) \) if and only if it is a relative boundary: \( \xi = \partial_{k+1} \eta + \gamma \) for some \( \eta \in C_{k+1}(X) \) and \( \gamma \in C_k(A) \). Fig. 2 is meant to clarify this concept.

**D. Combinatorial Laplacians**

The graph Laplacian [27] has various applications in image segmentation, graph embedding, dimensionality reduction for large data sets, machine learning, and more recently in consensus and agreement problems in distributed control of multi agent systems [18], [19]. For a graph \( G \), the Laplacian matrix is defined as \( L = BB^T \) where \( B \) is the vertex-by-edge-dimensional incidence matrix of \( G \). As it is evident from the definition, \( L \) is a positive semi-definite matrix. Also it is well-known that the Laplacian matrix can be written in terms of the adjacency and degree matrixes of \( G \) as well: \( L = D - A \), which implies that the \( i \)-th row of the Laplacian matrix

![Fig. 2. The 2-skeleton of a simplicial complex \( X \) and the subcomplex \( A \subset X \) consisting of all the boundary vertices and edges (heavy lines). Both \( \xi_1 \) and \( \xi_2 \) (highlighted) are relative 1-cycles, but only \( \xi_1 \) is the representative of a non-trivial element in \( H_1(X, A) \).](image-url)
matrix only depends on the local interactions between vertex \( i \) and its neighbors. The goal of this subsection is to present the generalization of this matrix to simplicial complexes and investigate its properties. The importance of these generalized Laplacian matrices (known as combinatorial Laplacians) lies in an observation made by Eckemann [28]; the fact that when working with real coefficients, the kernel of such a matrix spans a subspace isomorphic to the homologies.

The definitions and results of this subsection can be found in [26], [28].

**Definition 3:** Let \( X \) be a finite oriented simplicial complex. The \( k \)-th combinatorial Laplacian of \( X \) is the homomorphism \( \mathcal{L}_k : C_k(X) \to C_k(X) \) given by

\[
\mathcal{L}_k = \partial^*_k \circ \partial_k + \partial_{k+1} \circ \partial^*_k + 1
\]

(4)

where \( \partial^*_k \) is the adjoint of the operator \( \partial_k \) with respect to the inner product that makes the basis orthonormal.

The Laplacian operator, as defined above, is the sum of two positive semi-definite operators and therefore, any \( k \)-chain \( x \in \ker \mathcal{L}_k \) satisfies

\[
x \in \ker \partial_k \quad , \quad x \perp \text{im} \partial_{k+1}
\]

In other words, the kernel of the \( k \)-th combinatorial Laplacian consists of \( k \)-cycles which are orthogonal to the subspace \( \text{im} \partial_{k+1} \), and therefore, are not \( k \)-boundaries. This implies that the non-zero elements in the kernel of \( \mathcal{L}_k \) are representatives of the non-trivial equivalence classes of cycles in the \( k \)-th homology. This property was first observed by Eckmann [28] and is formalized in the following theorem [26].

**Theorem 1:** If the vector spaces \( C_k(X) \) are defined over \( \mathbb{R} \), then for all \( k \) there is an isomorphism

\[
H_k(X) \cong \ker \mathcal{L}_k
\]

(5)

where \( H_k(X) \) is the \( k \)-th homology of \( X \) and \( \mathcal{L}_k \) is its \( k \)-th combinatorial Laplacian. Moreover, there is an orthogonal direct sum decomposition of the vector space \( C_k(X) \) in the form of

\[
C_k(X) = \text{im} \partial_{k+1} \oplus \ker \mathcal{L}_k \oplus \text{im} \partial^*_k
\]

in which the first two summands comprise the set of \( k \)-cycles \( \ker \partial_k \), and the first summand is the set of \( k \)-boundaries.

The immediate implication of the above theorem is that the dimension of the subspace in the kernel of the \( k \)-th combinatorial Laplacian operator is equal to the \( k \)-th Betti number of
the simplicial complex. Before stating an example and clarifying the statement of Theorem 1, we would like to emphasize that, as stated earlier, for a finite simplicial complex, the boundary operators have matrix representations with respect to the bases of vector spaces $C_k(X)$. Therefore, one can use matrices to represent the combinatorial Laplacian operators in a similar manner: define the $k$-th combinatorial Laplacian matrix as

$$L_k = B_k^TB_k + B_{k+1}^TB_{k+1}^T \in \mathbb{R}^{n_k \times n_k}$$

where $B_k$ is the matrix representation of $\partial_k$ and $n_k$ is the number of $k$-simplices of $X$. Note that the expression for $L_0$ reduces to the well-known graph Laplacian matrix. Similarly, the combinatorial Laplacian matrices can be represented in terms of the adjacency and degree matrices [12] of the simplicial complex. More precisely, for $k > 0$,

$$L_k = D_u^{(k)} - A_u^{(k)} + (k + 1)I_{n_k} + A_l^{(k)},$$

where $A_u^{(k)}$ and $A_l^{(k)}$ are the upper and lower adjacency matrices, respectively and $D_u^{(k)}$ represents the upper degree matrix. (7) implies that the $i$-th row of $L_k$ only depends on the local interactions between $i$-th $k$-simplex and its upper and lower adjacent $k$-simplices. For simplicity, we use the matrix representation of the boundary and Laplacian operators for the rest of the paper.

**Example 1:** Consider the oriented simplicial complex depicted in Fig. 3, which consists of 6 vertices, 8 edges and 2 triangles. It is an easy exercise to show that the first combinatorial
Laplacian matrix is given by

\[
L_1 = \begin{pmatrix}
2 & -1 & 0 & 0 & 0 & -1 & 0 & 0 \\
-1 & 2 & -1 & 0 & 0 & 0 & -1 & -1 \\
0 & -1 & 3 & 0 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 3 & -1 & 0 & 0 & 0 \\
0 & 0 & 0 & -1 & 3 & -1 & 0 & 0 \\
-1 & 0 & 0 & 0 & -1 & 2 & 0 & -1 \\
0 & -1 & 0 & 0 & 0 & 0 & 4 & 0 \\
0 & -1 & 1 & 0 & 0 & -1 & 0 & 3
\end{pmatrix}
\]

where the edges are ordered as \([v_1v_2], [v_2v_3], [v_3v_4], [v_4v_5], [v_5v_6], [v_6v_1], [v_3v_5] \) and \([v_3v_6]\). The kernel of \(L_1\) is a one dimensional subspace spanned by the vector \(8 \ 8 \ 1 \ 1 \ 3 \ 8 \ 2 \ 5^T\). In Fig. 3, these values are depicted as flows on the edges of the simplicial complex. One can make a couple of observations based on the above computation: first of all, the dimension of \(\ker L_1\) is equal to the number of 1-dimensional holes in the simplicial complex, as was suggested by Theorem 1. Moreover, for any \(x \in \ker L_1\) the value of the algebraic sum of the flows entering each vertex is equal to zero. This is a consequence of the fact that any element in \(\ker L_1\) is also in \(\ker B_1\). Finally, note that the algebraic sum of the flows over any filled-in region is equal to zero as well. This is due to the fact that if \(x\) is in \(\ker L_1\), then \(B_2^T x = 0\) and therefore, \(x\) is orthogonal to \(\text{im} \ B_2\).

IV. DISTRIBUTED COVERAGE VERIFICATION IN THE ABSENCE OF LOCATION INFORMATION

One of the most prominent approaches for addressing the coverage problem has been the computational geometry approach, in which one uses the coordinates of the nodes and standard geometric tools (such as Delaunay triangulations or Voronoi diagrams) to determine coverage. These approaches often suffer from the drawback that they are too expensive to compute in real-time. Moreover, in most applications, they require exact knowledge of the locations of the sensing nodes. In general, due to their dependence on metric information, computational geometry approaches for coverage verification are not applicable, if the sensors are not equipped with localization devices.

In this section, we present a distributed coverage verification algorithm that can be used in the absence of any metric information. Unlike computational geometry approaches for coverage, this algorithm is based on computational algebraic topology which does not depend on location and...
orientation information. In essence, we compute the kernel of the first combinatorial Laplacian of a simplicial complex corresponding to the cover and use the fact that the first homology of the cover is trivial, if and only if the coverage is hole-free. The contents of this section are mainly from [10] and [12].

We first investigate the simplicial coverage framework: Let \( V = \{ v_1, \cdots, v_n \} \) denote the locations of \( n \) sensors deployed over a region \( D \subset \mathbb{R}^2 \), satisfying the assumptions presented in section II. These sensors are equipped with local coverage and communication capabilities, which enables them to exchange data with other sensors in their proximity: two sensors are capable of communicating with each other if the distance between them is less than or equal to \( r_b \). As for the coverage, we assume that any subset of the nodes which are in pairwise communication cover their entire convex hull. This implies that the total region covered is given by

\[
A(V) = \bigcup \{ \text{conv}(Q) | Q \subseteq V, \max_{v_i, v_j \in Q} \|v_i - v_j\|_2 \leq r_b \}.
\]

We are interested in verifying whether all the points within \( D \) are monitored by the sensors, i.e., whether \( D \subseteq A(V) \). Our assumptions of section II regarding the fence nodes guarantee that \( \partial D \subseteq A(V) \).

Since no location information is available to the sensors, we need to capture their communication and coverage properties combinatorially. For this purpose, we define what is known as the Vietoris-Rips complex corresponding to a given set points [29].

**Definition 4:** Given a set of points \( V = \{ v_1, \cdots, v_n \} \) in a finite dimensional Euclidean space and a fixed radius \( \epsilon \), the Vietoris-Rips complex of \( V, \mathcal{R}_\epsilon(V) \), is the abstract simplicial complex whose \( k \)-simplices correspond to unordered \((k+1)\)-tuples of points in \( V \) which are pairwise within Euclidean distance \( \epsilon \) of each other. Equivalently, the Rips complex is the flag complex of the proximity graph of \( V \), whose edges are pairs of points \( v_i, v_j \in V \) with \( \|v_i - v_j\| \leq \epsilon \).

Given this definition, one expects the Rips complex corresponding to the set of sensors to contain some information about the set \( A(V) \). In fact, the covered region \( A(V) \) is nothing but the image of the canonical projection map \( p : \mathcal{R}_\epsilon(V) \rightarrow \mathbb{R}^2 \) that maps each simplex in the Rips complex affinely onto the convex hull of its vertices in \( \mathbb{R}^2 \), known as the Rips shadow. The following theorem due to Chambers et. al [23] indicates that the Rips complex is rich enough to contain the required topological and geometric properties of its shadow.

**Theorem 2:** Let \( V \) denote a finite set of points in the plane, with the corresponding Rips
complex $\mathcal{R}_e(V)$. Then the induced homomorphism $p_* : \pi_1(\mathcal{R}_e(V)) \to \pi_1(A(V))$ between the fundamental groups of the Rips complex and its shadow is an isomorphism.

Equivalently, Theorem 2 states that a cycle $\gamma$ in the Rips complex is contractible if and only if its projection $p(\gamma)$ is contractible in the Rips shadow [22]. The important implication of this theorem is that the first homology groups of the Rips complex and its shadow are isomorphic as well. Therefore, the triviality of the first homology of the Rips complex provides a necessary and sufficient condition for a hole-free coverage of $D$.

In addition to the above, the Rips complex has the desirable property that it can be easily formed just by using communication among nearest neighbors. This is due to the fact that the Rips complex is the flag complex of the proximity graph and as a result, solely depends on connectivity information. This property makes the Rips complex a desirable combinatorial abstraction of the sensor network, which can be used for distributed coverage verification in the absence of location information. Also, as stated in the previous section, the combinatorial Laplacians carry valuable information about the topological properties of a simplicial complex. In particular, $\ker L_1(\mathcal{R}_{rb}) = \{0\}$ guarantees that $H_1(\mathcal{R}_{rb})$ is trivial and as a result, all the 1-cycles over the Rips complex are null-homologous. Therefore, based on Theorem 2, $\ker L_1(\mathcal{R}_{rb}) = \{0\}$ serves as a necessary and sufficient condition for the Rips shadow to be hole-free. One way to compute a generic element in the kernel of the Laplacian matrix is through the dynamical system $\dot{x}(t) = -L_1x(t)$ which asymptotically converges to such an element. This implies the following theorem which was first stated and proved in [12].

**Theorem 3:** The linear dynamical system

$$\dot{x}(t) = -L_1x(t), \quad x(0) = x_0 \in \mathbb{R}^{n_1}$$

(8)

is globally asymptotically stable if and only if $H_1(\mathcal{R}) = 0$, where $x(t)$ is a vector of dimension $n_1$ (the number of 1-simplices of the simplicial complex) and $L_1$ is the first combinatorial Laplacian matrix of the Rips complex $\mathcal{R}_{rb}$.

Note that for any initial condition $x(0)$, the trajectory $x(t); t \geq 0$ always converges to a point in $\ker L_1$. Thus the asymptotic stability of the system is an indicator of an underlying trivial homology. In different terms, since $x^* = \lim_{t \to \infty} x(t)$ is an element in the null space of $L_1$, it is a representative of a homology class of the Rips complex. Clearly, if $x^* = 0$ for all initial conditions, then the first homology of the simplicial complex consists of only a trivial class and
therefore, the simplicial complex is hole-free.

The importance of using the first combinatorial Laplacian of the simplicial complex is not limited to the above theorem. Its very specific structure guarantees that the update equation (8) is effectively a local update rule. In fact, this update rule works in the spirit of a certain class of distributed algorithms known as gossip algorithms [30], whereby the local state value of an edge is updated using estimates from edges that are lower adjacent to it. The reader may also note the connection between the distributed update (8) and the distributed, continuous-time consensus algorithms, in which the graph Laplacian is used in order to reach a consensus (a point in the kernel) over a connected graph [19].

In summary, in order to verify coverage in a network of fixed sensors, it is sufficient to setup the distributed linear dynamical system (8) for a random initial condition and observe the asymptotic state value as \( t \to \infty \). If this distributed dynamical system converges to zero, then the first Betti number of the Rips complex is zero, and therefore, the Rips shadow (which is the actual region covered by the sensors) is hole-free. Conversely, if the asymptotic value of (8) is non-zero, then the first homology of the Rips complex is non-trivial and therefore, Theorem 2 implies the existence of a non-contractible 1-cycle in the Rips shadow and hence, coverage holes. Note that our assumption regarding the existence of a set of fence nodes located on the boundary of \( D \) is crucial in avoiding boundary effects. These fence nodes guarantee that if there exists a coverage hole, it is located in the interior of the domain \( D \).

We now consider the symmetric coverage framework, in which each sensor is capable of covering a disk of radius \( r_c \) and communicate with other sensors within distance \( r_b \leq r_c \sqrt{3} \). In this case, the total region covered by the sensors is the union of disks of radius \( r_c \) centered at the location of the sensors: \( U(V) = \bigcup_{v_i \in V}\{x \in \mathbb{R}^2 : \|x - v_i\| \leq r_c\} \). Similar to the previous framework, we define a combinatorial object, known as the Čech or Nerve complex, that captures the topological properties of the set \( U(V) \).

**Definition 5:** Given a finite collection of disks \( \{U_v : v \in V\} \) with radius \( \epsilon \) centered at points \( v_i \), the Čech complex of the collection denoted by \( C_\epsilon(V) \) is the abstract simplicial complex whose \( k \)-simplices correspond to non-empty intersections of \( k + 1 \) distinct elements of \( \{U_v : v \in V\} \).

In other words, this complex is simply formed by associating a vertex to each disk, and then

---

\(^4\)This assumption on strong degree of control along the boundary is not strictly required and can be relaxed [10].
adding edges and other higher order simplices based on the overlap of the disks. The following theorem, known as the Čech Theorem or the Nerve Lemma, indicates why the Čech complex captures the topological properties of the region covered by the sensor footprints [31].

Theorem 4: Given a finite collection of disks $U_v$ centered at points $v \in V$, the Čech complex $C_\epsilon(V)$ has the homotopy type of the union of the disks in the collection, $\bigcup_{v \in V} U_v$.\(^5\)

The above theorem implies that the Čech complex carries the same homological properties of the union of the sets. In particular, both objects have isomorphic homologies in all dimensions.\(^6\) Therefore, in order to verify coverage in a given domain by a set of sensors, one only needs to look at the homologies of the underlying Čech complex. If this simplicial complex has no holes, neither does the sensor cover. However, computation of this simplicial complex and hence, its homologies is not an easy task, as it requires localization of each sensor as well as distance measurements in order to verify overlaps of footprints. Furthermore, as shown in [8], the Čech complex is very fragile with respect to uncertainties in distance and location measurements. In the absence of location information, an alternative would be to use the Rips complex instead, which can be formed uniquely from the communication graph of the network. Unfortunately, the Rips complex is not rich enough to contain all the topological and geometric information of the Čech complex and in general does not provide much information about coverage holes. Recently, it is shown that in certain cases, the Rips complex does in fact carry the necessary information to extract the homological properties of the cover. Namely, the authors of [10] have shown that a Rips complex with parameter $\epsilon$, $R_\epsilon$, is a subcomplex of a Čech complex corresponding to disks of radius $\epsilon \sqrt{3}$ centered at the location of vertices of the Rips complex. As a result, our assumption of $r_b \leq r_c \sqrt{3}$ leads to

$$R_{r_b} \subseteq C_{r_c} \left( \{U_v : v \in V \} \right),$$

which implies $A(V) \subseteq U(V)$, where $A(V)$ is the shadow of the Rips complex with parameter $r_b$ and $U(V)$ is the actual region covered by the sensors in the symmetric coverage framework with

\(^5\)The statement of this theorem holds for any collection of contractible sets when all nonempty intersections of all subcollections are contractible.

\(^6\)Note that homotopy equivalence of two topological spaces is much stronger than having isomorphic fundamental groups, as was the case in Theorem 2. In other words, the Čech complex tells us much more about the union of disks than the Rips complex does about its shadow.
coverage radius $r_c$. Hence, if the Rips complex with broadcast disks of radius $r_b$ is hole-free, then so is the sensor coverage. This result would serve as a sufficient homological criterion for coverage verification. Note that the case of $r_b = \sqrt{3}r_c$ corresponds to the tightest such sufficient condition for planar networks.

In summary, in order to verify a successful coverage in a distributed fashion, the sensors need to compute the first homology of the Rips complex $\mathcal{R}_{r_b}$ using the local neighborhood information available to them. The triviality of the first homology of this simplicial complex provides a sufficient condition for a hole-free coverage of the region of interest $D$. Therefore, once again, one can set up the linear dynamical system (8) corresponding to the Rips complex with parameter $r_b$ and observe its asymptotic behavior. Similar to the simplicial coverage framework, the asymptotic stability of this dynamical system is a sufficient condition for a hole-free coverage, although it is not necessary anymore.

As a last remark note that (8) is an edge-dimensional dynamical system, where each element of the vector $x(t)$ corresponds to a 1-simplex. However, in both frameworks, edges and all other higher order simplices are simply combinatorial objects and the only real physical entities capable of computation are the sensors themselves. Therefore, in order to implement (8) in a sensor network one needs a protocol to assign the computation required by an edge to its adjacent nodes. One such algorithm is presented in [13], in which the authors obtain a local representation of the Rips complex and implement the dynamical system in Theorem 3 at the node level.

V. Hole Localization: Distributed Computation of the Sparsest Generator

In the previous section, we presented a coverage verification algorithm for a sensor network in which the nodes have no location or distance information. As noted before, this distributed algorithm is based on the close topological relationship between the actual cover and the Rips complex as its combinatorial representation. Unfortunately, this verification algorithm is not powerful enough to provide any further information on the cover. All it is capable of is verifying whether the coverage is successful (hole-free) or not. In most practical scenarios, one’s interest is not simply limited to coverage verification. In fact, we are as much interested in the location, number or the size of the coverage holes (if they exist). Therefore, the algorithm of section III needs to be followed by an algorithm which can reveal further information about the cover.

In this section, we present a distributed algorithm which is capable of “localizing” coverage...
holes in a sensor network with no location or metric information. By hole localization, we mean detecting cycles over the proximity graph of the network that encircle the coverage holes. The tightest of such cycles provides information on the location and the size of the hole in the Rips shadow.\footnote{Note that in the simplicial framework, the Rips shadow coincides with the actual cover, whereas in the symmetric framework it is only a subset of the region covered by the sensors.} Similar to the previous algorithm, the results of this section are also based on the algebraic topological invariants, namely the homology, of the cover and the Rips complex of the network. In essence, in order to find the coverage holes, our algorithm computes the sparsest generator of a non-trivial class of homologous 1-cycles in the first homology of the simplicial complex, which corresponds to the shortest possible cycle around the holes. Our method is more general than the algorithms presented in [20], [21], where it is explicitly assumed that the simplicial complex is embedded on an orientable surface. It is also different from the results in [22] in the sense that it is not limited to Rips complexes, is distributed in nature, and does not use node coordinates.

Before presenting the algorithm, we state a few remarks regarding the relationship between the sparsest generator of the homology classes and the location of the holes. It is important to keep in mind that we are using simplicial complexes which are combinatorial objects. Therefore, for hole localization in the absence of metric information, the best we can hope for is computing the shortest cycle encircling a hole, which is also a combinatorial object. For instance, consider the two different sensor configurations and the region covered by them as depicted in Fig. 4. Although the region covered is different, they are combinatorially equivalent as far as the Rips complex is concerned. Therefore, in both cases, any hole localization algorithm leads to the same result.

Another case that is worth mentioning is the case that the simplicial complex contains multiple
holes. It is quite possible that in the case that two holes are close relative to their sizes, the sparsest generator of the homology class encircles both of them simultaneously, rather than each hole individually. Fig. 5 is meant to clarify this case. In either case, the sparsest 1-cycle provides valuable information on the location and sizes of the holes.

With the above in mind, we present an algorithm which is capable of finding the shortest non-trivial cycle in a homology class. Intuitively, given a representative cycle of a non-trivial homology class, our algorithm is capable of computing the sparsest generator of that homology class in a distributed fashion, simply by removing components corresponding to contractible cycles and “tightening” it around the holes. Therefore, in order to find the shortest cycle in a homology class, the algorithm needs an initial non-trivial 1-cycle in that class. Clearly, any non-zero point in \( \ker L_1 \) can potentially serve as such an initial 1-cycle. The immediate advantage of using \( x \in \ker L_1 \) is that one can easily compute such a point in a distributed manner as the limit of linear dynamical system (8). The following example clarifies the idea behind the algorithm.

**Example 2:** Consider the 2-dimensional simplicial complex depicted in Fig. 3. As was shown in Example 1, the kernel of the first combinatorial Laplacian of this complex is one-dimensional. Therefore, the distributed linear dynamical system (8) converges to a non-zero vector in the span of \( \begin{bmatrix} 8 & 8 & 1 & 1 & 3 & 8 & 2 & 5 \end{bmatrix}^T \) for almost all initial conditions. Notice that all the edges, including

---

8By terms such as close or big, we simply mean combinatorially close (in terms of hop count) and combinatorially big (in terms of the length of the shortest cycle).
edges \([v_3v_4], [v_3v_5], [v_4v_5]\) and \([v_5v_6]\) that are not adjacent to the hole, have non-zero values asymptotically. In other words, no element of \(\ker L_1\) is “tight” around the hole of the simplicial complex. Another key observation is that any \(x \in \ker L_1\) can be written as a linear combination of the three fundamental cycles in the 1-skeleton of the simplicial complex:

\[
x = 8\alpha c_1 + 3\alpha c_2 + \alpha c_3
\]

where

\[
c_1 = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 1 & 0 & 1 \end{bmatrix}^T
\]

\[
c_2 = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 0 & 1 & -1 \end{bmatrix}^T
\]

\[
c_3 = \begin{bmatrix} 0 & 0 & 1 & 1 & 0 & 0 & -1 & 0 \end{bmatrix}^T
\]

and \(\alpha\) is some real number. Among these cycles, only the first one corresponds to the hole, while the other two are simply contractible cycles corresponding to boundaries of 2-simplices. Therefore, in order to find a tight cycle around the hole, one needs to subtract the right amount of null-homologous 1-cycles encircling 2-simplices (in this example \(3\alpha\) and \(\alpha\) respectively) from \(x\). What remains is simply a 1-cycle with non-zero values only over the edges that are adjacent to the hole. Note that this cycle is also the sparsest generator of the non-trivial element of the first homology of the simplicial complex.

Computing the tightest cycle around the hole in the above example is simple, due to the fact that the simplicial complex only consists of very few simplices. Unfortunately, once the simplicial complex becomes large, it is not an easy task to compute the right amount of contractible cycles to subtract from an element in \(\ker L_1\), and find a sparse representative of the homology class. Moreover, in the absence of a centralized scheme, it is reasonable to assume that the elements of the vector \(x \in \ker L_1\) are only known locally to the nodes. In fact, this is the case if the kernel element is computed in a distributed fashion using the dynamical system (8). Therefore, we need an algorithm which is capable of finding the sparsest non-trivial generator of the homology classes of a simplicial complex by using only local information.

A. Computing the Sparsest Generator: IP Formulation

Consider a simplicial complex \(X\) with the first combinatorial Laplacian \(L_1\). By construction, any element in the null space of \(L_1\) is a 1-cycle that is orthogonal to the subspace spanned by
the boundaries of the 2-simplices. In other words, \( x \in \ker L_1 \subset \mathbb{R}^{n_1} \) implies \( x \in \ker B_1 \) and \( x \perp \operatorname{im} B_2 \). Therefore, as stated in section III, any non-zero \( x \) in the kernel of the first combinatorial Laplacian is a representative element of a non-trivial homology class of \( X \). However, as in Example 2, \( x \) is not necessarily the sparsest representative of the homology class it belongs to. In general, given a generator \( x \) of a homology class, the sparsest generator of that class can be computed as the solution to the following integer programming optimization problem:

\[
\begin{align*}
\text{Minimize} & \quad \| y \|_0 \\
\text{subject to} & \quad y = x + B_2 z
\end{align*}
\]  

(9)

where \( \| \cdot \|_0 \) is the \( \ell_0 \)-norm of a vector, equal to the number of non-zero elements of that vector, and \( B_2 \) is the matrix representation of the second boundary operator \( \partial_2 \). Note that if \( x \) is a 1-cycle, then the minimizer \( y^* \) is also a 1-cycle in the kernel of \( B_1 \). Moreover, the constraint \( y - x \in \operatorname{im} B_2 \) guarantees that both \( x \) and \( y^* \) are representatives of the same homology class, or in other words, adding and subtracting null-homologous cycles does not change the homology class. Therefore, any solution of the above optimization problem is the sparsest generator of the homology class that \( x \) belongs to, and has the desired property that it is the tightest possible cycle (in terms of the length) around the holes represented by that homology class.

B. LP Relaxation

The optimization problem (9) has a very simple formulation. However, due to the 0-1 combinatorial element in the problem statement, solving it is not, in general, computationally tractable. In fact, in [32] the authors show that computing the sparsest generator of an arbitrary homology class is NP-hard.

A popular relaxation for solving such a problem is to minimize the \( \ell_1 \)-norm of the objective function rather than its \( \ell_0 \)-norm [33]:

\[
\begin{align*}
\text{Minimize} & \quad \| y \|_1 \\
\text{subject to} & \quad y = x + B_2 z
\end{align*}
\]  

(10)

This relaxation\(^9\) is a linear programming (LP) problem and can be solved quite efficiently. An argument similar to before shows that the minimizer of the above optimization problem is also

\(^9\)Note that (10) is not exactly a relaxation of (9), as the two problems have the same feasibility sets. However, one can show that there exists an LP equivalent to (10) which is a relaxation of an IP equivalent to (9).
a 1-cycle homologous to the initial $x$, since their difference is simply a null-homologous cycle in the image of $B_2$.

In general, due to the relaxation, the minimizer of (10) is simply an approximation to the minimizer of (9) and has a larger $\ell_0$-norm. However, in certain cases the solutions of the two problems coincide. In the next theorem, we present conditions under which the two minimizers have the same zero/nonzero pattern. Under such conditions, we would be able to compute the sparest generator of the homology class of $x$ efficiently.

**Theorem 5:** Suppose $X$ is a simplicial complex with first combinatorial Laplacian $L_1$, and consider the non-trivial generator $x \in \ker L_1$. Also suppose that the sparest generator of any homology class is unique and is a linear combination of the shortest cycles that encircle the holes represented by that class. Then, the minimizers of the problems (9) and (10) coincide.

**Proof:** See the Appendix.

The above theorem states that, under the given conditions, the $\ell_1$ minimizer is the sparest generator of its homology class as well, and therefore, its non-zero entries indicate the edges of the 1-cycle that are tight around the holes. As a consequence, one can compute this sparse generator efficiently, using methods known for solving LPs.

**Remark 1:** Theorem 5 states that a necessary condition for equality of the two minimizer is the uniqueness of the sparest generator of each homology class. When the $\ell_0$ minimizer is not unique, not only every $\ell_0$ minimizer is a solution to (10), but so is any convex combination of those minimizers. This is due to the fact that if two vectors have the same $\ell_1$-norm, then so does any vector in their convex hull. In such cases, solving (10) results in a 1-cycle in the convex hull of the minimizers of (9). The proof of this more general statement is similar.

**Remark 2:** The condition of Theorem 5, restated mathematically in the Appendix, is also worth considering. One very important case for which the condition holds, is the case that the simplicial complex has only one hole. Another is the case that the holes in the simplicial complex are far from each other relative to their sizes. In either case, the shortest representative cycle of any homology class is simply a linear combination of the shortest cycles encircling the holes separately. It is important to note that even when the condition does not hold, the solution of (10) is a relatively sparse (although not necessarily the sparest) 1-cycle, and therefore, can be used as a good approximation to localize the holes.
C. Decentralized Computation: The Subgradient Method

As mentioned before, unlike the original IP problem (9), one can convert (10) to a linear programming problem and solve it efficiently using methods known for solving LPs. However, applying the subgradient method [34], [35] enables us to compute the ℓ_1 minimizer in a distributed manner. Although the convergence would be slower than usual methods for solving linear programs, the added value of decentralization makes the method worthwhile.

One can rewrite the optimization problem (10) as

$$\text{Minimize } \|x + B_2 z\|_1$$

where \(n_2\) is the number of the 2-simplices of the simplicial complex. A subgradient for the objective function in the above problem is the sign function. Therefore, the subgradient update can be written as

$$z^{(k+1)} = z^{(k)} - \alpha_k B_2^T \text{sgn}(B_2 z^{(k)} + x)$$

with the initial condition \(z^{(0)} = 0\). Note that \(z\) is a face-dimensional vector and the iteration updates an evaluation on the 2-simplices of the simplicial complex. The most important characteristic of (12) is that, due to the local structure of \(B_2\), it can be implemented in a distributed manner, if the initial \(x\) is known locally. By picking a small enough constant step size \(\alpha_k\), it is guaranteed that the update (12) gets arbitrarily close to the optimal value [34]. By choosing more sophisticated dynamic step sizes we can improve the convergence properties of the above algorithm to the optimal solution, which under the conditions of Theorem 5 is the sparsest generator (or a convex combination of the sparsest generators) of the homology class of the initial 1-cycle \(x\). In section VIII we provide non-trivial simulations of this algorithm.

VI. DISTRIBUTED DETECTION OF REDUNDANT SENSORS

In the previous sections we presented a homological criterion for coverage. Namely, based on the results of [9], we argued that a sufficient condition for a successful coverage is to have no holes in the flag complex of the proximity graph, i.e., the Rips complex of the network. This condition is translated into algebraic topological terms as \(H_1(\mathcal{R}_{rb}) = 0\), or, that any cycle in the communication graph can be realized as the boundary of a surface built from the 2-simplices of \(\mathcal{R}_{rb}\). Furthermore, we showed that based on the space decomposition theorem (Theorem 1), the
first combinatorial Laplacian can be used to verify the above mentioned homological criterion in a distributed manner.

In this section, we present a distributed algorithm which is capable of computing a sparse cover of the domain $D$ and detect redundancies in the sensor network, in the absence of location information. In other words, the algorithm enables us to “turn off” redundant sensors without impinging upon the coverage integrity. As before, we formulate the problem of finding a sparse cover as an optimization problem to compute the sparsest generator of a certain homology class, and use subgradient methods to solve it in a distributed way. However, in contrast to the previous sections, we use the second homology of the Rips complex relative to its boundary, rather than its first homology. The advantage of the second relative homology lies in the fact that it is more robust to extensions and therefore, yields stronger information about the actual cover \cite{10}.

Consider the Rips complex $\mathcal{R}$ corresponding to network of the sensors deployed over region $D$ and consider $\mathcal{F} \subset \mathcal{R}$ to be the subcomplex that is canonically identified with the fence nodes over $\partial D$. If this cycle is null-homologous - that is, if $[\mathcal{F}] = 0$ in $H_1(\mathcal{R})$ - then, the coverage is hole-free. In such a case, there exists a 2-chain which bounds $\mathcal{F}$:

$$\forall 1\text{-cycle } \beta \in C_1(\mathcal{F}), \quad \exists \alpha \in C_2(\mathcal{R}) \text{ s.t. } \beta = \partial_2 \alpha$$

Therefore, when translated into the language of algebraic topology, such a 2-chain $\alpha$, which is not necessarily unique, is a relative 2-dimensional homology class, a certain generator in $H_2(\mathcal{R}, \mathcal{F})$.

As a result, the condition for a hole-free successful coverage can be rewritten in terms of the second relative homology classes:

**Theorem 6:** For a set of nodes $V$ in a domain $D \subset \mathbb{R}^2$ satisfying the assumptions of section II, the sensor cover contains $D$ if there exists $[\alpha] \in H_2(\mathcal{R}, \mathcal{F})$ such that $\partial_2 \alpha \neq 0$.

The proof of the above theorem can be found in \cite{10}. Intuitively, the 2-chain $\alpha$ has the appearance of “filling in” $D$ with triangles composed of projected 2-simplices from $\mathcal{R}$. Note that the relative group $H_2(\mathcal{R}, \mathcal{F})$ captures the second homology of the quotient space $\mathcal{R}/\mathcal{F}$, in which all the simplices in $\mathcal{F}$ are identified. This can be done by adding a “super node” to the complex, as depicted in Fig. 6. If the Rips complex does not have any hole, then the topology of this quotient space is that of a sphere, and therefore, the relative homology $H_2(\mathcal{R}, \mathcal{F})$ has a non-trivial generator. On the other hand, if the subcomplex $\mathcal{F}$ is not contractible, then the relative homology has no generator with non-zero values on the boundary.
Fig. 6. If the first homology of $R$ is non-trivial, then the second relative homology $H_2(R, F)$ has no generator with values on the boundary. Conversely, if the second homology relative to the boundary has a non-trivial generator, then $H_1(R) = 0$.

Note that the dimension of the second relative homology $H_2(R, F)$ may be greater than one. This can happen if there exists a 2-cycle which is a generator of $H_2(R)$ as well as $H_2(R, F)$, as depicted in Fig. 7. Such 2-cycles do not represent a true relative class, as they may still exist even if the fence cycle $F$ is not contractible. Hence, Theorem 6 requires the existence of a relative 2-cycle $\alpha$ with a non-zero boundary.

Given the above, the reader may have noticed that the minimal cover is simply the sparsest generator of a second homology class of $R$ relative to $F$. Therefore, one can formulate the problem of finding the sparsest cover over $D$ as an optimization problem, simply as a generalization of the results of the previous section to a higher dimension. The only difference lies in the fact that instead of the Rips complex corresponding to the network, we use the quotient complex $\overline{R} = R / F$ which is obtained by identifying all the simplices of $F$ with a supper node. Once this quotient simplicial complex is formed\(^{10}\), we compute its second combinatorial Laplacian in a distributed manner, and by running the decentralized linear dynamical system $\dot{x}(t) = -L_2 x(t)$ obtain a point $x \in \ker L_2$ asymptotically. The limit of this dynamical update is a relative 2-cycle which does not vanish on the boundary almost surely. Once such a 2-cycle $x$ is computed, the minimizer of the optimization problem

$$\begin{align}
\text{Minimize} & \quad \|y\|_0 \\
\text{subject to} & \quad y = x + B_3 z
\end{align}$$

represents the sparsest generator of the relative homology class that $x$ belong to. In the above problem, $B_3$ is the triangle-by-tetrahedron incidence matrix of the quotient complex $\overline{R} / F$, $x$ and $y$ are 2-cycles and $z$ is a 3-chain. Similar to problem (9), the constraint $y - x \in \text{im } B_3$.

\(^{10}\)Note that this object can be formed in a distributed fashion. All that is required is that the fence nodes take the local neighborhood relations of each other into account and update their values together.
The eight faces of the octahedron form a non-trivial 2-cycle $\alpha$ such that $[\alpha] \in H_2(\mathbb{R})$. However, $\alpha$ has a vanishing boundary $\partial_2 \alpha = 0$, and therefore, does not correspond to a true relative 2-cycle.

guarantees that $y$ and $x$ are homologous 2-cycles. Not surprisingly, one can prove a result similar to Theorem 5 for (13) and its relaxation

$$\text{Minimize} \quad \|x + B_3 z\|_1$$

which can be solved by the means of the distributed subgradient update

$$z^{(k+1)} = z^{(k)} - \alpha_k B_3^T \text{sgn}(B_3 z^{(k)} + x).$$

The above distributed iteration would lead to a sparse generator of the second relative homology, in which most of the 2-simplices have a corresponding value equal to zero. In the optimal solution, if all the 2-simplices that contain a vertex have value zero, that vertex and all its communication links can be removed from the network, without generating a coverage hole.

VII. SIMULATIONS

In this section, we present the simulation results for the algorithms presented in sections V and VI, for hole localization and computation of the minimal cover, respectively.

A. Hole Localization

We demonstrate the performance of our distributed hole localization algorithm with a randomly generated numerical example. Fig. 8(a) depicts the Rips shadow of a simplicial complex on $n = 81$ vertices distributed over $\mathbb{R}^2$. The 2-skeleton of this simplicial complex consists of 81 vertices, 372 edges, and 66 faces (2-simplices). As expected from Fig. 8(a), the null space of the first combinatorial Laplacian of this Rips complex has dimension 2. The two non-trivial homology classes correspond to two zero eigenvectors of the Laplacian matrix. We generated a point in $x \in \ker L_1$ by running the distributed linear dynamical system (8) with a random...
initial condition $x(0)$. The edge-evaluation of the limiting $x \in \ker L_1$ is depicted in Fig. 8(b), where the thickness of an edge is directly proportional to the magnitude of its corresponding component in $x$. It can be seen that for this 1-cycle in the null space of $L_1$, all the components more or less have the same order of magnitude. In order to localize the two holes, we ran the subgradient update (12) with a diminishing square summable but not summable step size. The edge evaluation of the 1-cycles after 1000 and 4000 iterations are depicted in Figs. 7(c) and (d). These figures illustrate that after enough iterations, the subgradient method converges to a 1-cycle that has non-zero values only over the cycles that are tight around the holes. Therefore, the algorithm is capable of localizing the coverage holes. In Fig. 8(d), the value of the 12 edges adjacent to the holes are 3 orders of magnitude higher than all the others.

Note that our algorithm is only capable of finding the tightest minimal-length cycles surrounding the holes, which do not necessarily coincide with the cycles that are closer distance-
Finding the minimal generator of the second relative homology $H_2(\mathcal{R}, \mathcal{F})$ leads to a minimal cover. 32 of the sensors can be turned off without generating any coverage holes.

wise to the holes. As stated before, after all, we are not using any metric information and the combinatorial relations between vertices is the only information available. Moreover, in case there are two minimal-length cycles surrounding the same hole (as in the upper hole in Fig. 7), then any convex combination of those is also a minimizer to the LP relaxation problem (10). In such cases, the subgradient method in general converges to a point in the convex hull of the two solutions, rather than a corner solution. Also note that the two holes in the Rips complex are far relative to their sizes and therefore, Theorem 5 guarantees that the solution obtained by the $\ell_1$ minimization lies in the convex hull of the $\ell_0$ minimizers.

**B. Computing a Sparse Cover**

Fig. 9 illustrates the performance of the algorithm presented in section VI. The randomly generated Rips complex used for this simulation is made up of 66 vertices, 22 of which function as fence nodes (Fig. 9(a)). The second relative homology of this simplicial complex consists of only one non-trivial class of relative 2-cycles. In order to compute a non-trivial representative of the second relative homology, we introduced an extra super node, connected to all the fence nodes. We computed the second combinatorial Laplacian of the resulting complex and used the linear update $\dot{x}(t) = -L_2 x(t)$ to obtain a point in the null space of $L_2$. Subgradient update (15) is used to solve the optimization problem (13). The minimizer 2-cycle is depicted in Fig 9(b). Any vertex which does not belong to a 2-simplex with a non-zero evaluation at the optimal can be removed, without impinging upon the coverage integrity. As illustrated in Fig. 9(b), 32 of
the vertices can be removed, while the Rips shadow remains hole-free.

As a last remark, note that either one of the vertices \( a \) or \( b \) in Fig. 9(b) can also be removed from the network without generating a coverage hole. In fact, the removal of either one, would lead to even a sparser solution than the one obtained by the subgradient update. This is due to the fact that the generator depicted in Fig. 9(b) is in fact in the convex hull of two distinct solutions of the original integer programming problem (13), and as earlier, if the original problem has more than one minimizer, then any convex combination of them is also a minimizer of the LP relaxation problem (14).

VIII. Conclusions

In this paper, we presented distributed algorithms for coverage verification in a sensor network, when no metric information is available. We used the simplicial complexes and their combinatorial Laplacians in order to abstract away the topological properties of the network. Furthermore, we showed how the simplicial homologies of the Rips complex can provide information about the cover, in the absence of location information. In particular, we illustrated the relationship between the kernel of the first combinatorial Laplacian of the Rips complex and the number of coverage holes. Moreover, we formulated the problem of localizing the coverage holes (in the sense of finding the tightest 1-cycle encircling them) as an optimization problem and used subgradient methods to solve it in a distributed fashioned. Along the same lines, we showed that how by generalizing the same argument to a higher dimension one can compute the sparsest cover, and therefore detect redundancies in the network. In particular, we presented a subgradient update that is capable of computing the sparsest generator of the second homology classes of the Rips complex relative to the fence nodes, in a decentralized manner. We then used the minimizer to detect redundant sensors. The algorithms presented in this paper can be extended to compute the sparsest generator of any higher dimension homology class as well. Finally, we provided non-trivial simulations to demonstrate the performance of our algorithms.

Appendix

Consider an oriented simplicial complex \( X \) with the first Betti number \( b \), where the holes are labeled 1 through \( b \). By \( h(\alpha_1, \ldots, \alpha_b) \) we denote the class of homologous 1-cycles that encircle
the \(i\)-th hole \(\alpha_i\) many times in a given direction. We also assume that the shortest representative cycle that encircles one single hole is unique and is denoted by \(c_i^*\). In other words,

\[
c_i^* = \arg \min_{c} \|c\|_0 \quad \text{s.t. } \quad c \in h(e_i)
\]

where \(e_i\) is the \(i\)-th coordinate vector. Since \(c_i^*\) is the sparsest 1-cycle that encircles the \(i\)-th hole only once, we have the following lemma.

**Lemma 2:** \(c_i^*\) is 1-cycle which only has value in \(\{0, 1, -1\}\).

We now restate and prove Theorem 5.

**Theorem 5:** Given a simplicial complex \(X\), suppose that \(\arg \min_{c \in h(\alpha)} \|c\|_0 = \sum_{i=1}^{b} \alpha_i c_i^*\), for all \(\alpha \in \mathbb{R}^b\). Then, for all \(\alpha \in \mathbb{R}^b\) we have,

\[
\arg \min_{c \in h(\alpha)} \|c\|_0 = \arg \min_{c \in h(\alpha)} \|c\|_1.
\]

**Proof:** First we prove that the two minimizers have the same zero/non-zero pattern. Given a class \(h(\alpha)\), suppose that the \(\ell_1\) minimizer denoted by \(y\) does not have the same pattern as the \(\ell_0\) minimizer. This means that there exists an edge \(\sigma_1\) in the simplicial complex such that \(y\) has a positive value on, but the \(\ell_0\) minimizer does not. Since \(y\) is a 1-cycle, there exists another edge \(\sigma_2\) lower-adjacent to \(\sigma_1\) with a non-zero value. Without loss of generality, we assume that the directions are defined such that all the values are positive. Reapplying the same argument implies that \(\sigma_1\) belongs to a set \(E\) of edges, all with positive values and forming a simple loop over the simplicial complex. Moreover, it implies that \(\tilde{c}_j = \mathbb{I}_{\{\sigma_j \in E\}}\) is a 1-cycle. Note that \(\tilde{c}\) is a simple 1-cycle which only takes values in \(\{0, 1, -1\}\). Finally, define \(\gamma > 0\) to be the smallest value that the edges in \(E\) take in the \(\ell_1\) minimizer \(y\).

The 1-cycle \(\tilde{c}\) belongs to some homology class \(h(\mu)\), that is, the class of 1-cycles that encircle the \(i\)-th hole \(\mu_i\) many times. Without loss of generality, we can assume that \(\mu_i \geq 0\) for all \(0 \leq i \leq b\). Define \(y' = y - \gamma \tilde{c} + \gamma (\sum_{i=1}^{b} \mu_i c_i^*)\) for which we have,

\[
\|y'\|_1 \leq \|y - \gamma \tilde{c}\|_1 + \gamma \sum_{i=1}^{b} \mu_i \|c_i^*\|_1
\]

\[
= \|y\|_1 - \gamma \|\tilde{c}\|_1 + \gamma \sum_{i=1}^{b} \mu_i \|c_i^*\|_1
\]

\[
= \|y\|_1 - \gamma \|\tilde{c}\|_0 + \gamma \sum_{i=1}^{b} \mu_i \|c_i^*\|_0 < \|y\|_1,
\]
The first inequality is a consequence of the triangular inequality. The following equality is due to the fact that we assumed that $\gamma$ is the smallest value on the edges of $\tilde{c}$ at $y$. In the next equality, we use that fact that $\tilde{c}$ and all $c_i^*$ are 1-cycles with values in $\{0, 1, -1\}$, which means that their $\ell_1$ and $\ell_0$ norms are equal. Finally, the last inequality is due to assumption of the theorem.

In summary, there exists a 1-cycle $y'$ homologous to $y$ with a smaller $\ell_1$-norm, which contradicts the fact that $y$ is the $\ell_1$ minimizer. Therefore, $\arg \min_{c \in h(\alpha)} \|c\|_0$ and $\arg \min_{c \in h(\alpha)} \|c\|_1$ have the same zero/non-zero pattern for all $\alpha$. Also note that both minimizers belong to the same homology class $h(\alpha)$. As a result, the two must be equal.

ACKNOWLEDGMENT

The authors would like to thank Robert Ghrist for many helpful comments and discussions.

REFERENCES


