HAL
open science

Reduction Algorithm for Simplicial Complexes

Anais Vergne, Laurent Decreusefond, Philippe Martins

To cite this version:

Anais Vergne, Laurent Decreusefond, Philippe Martins. Reduction Algorithm for Simplicial Complexes. INFOCOM 2013, Apr 2013, Turin, Italy. 10.1109/INFCOM.2013.6566818 . hal-00688919v2

HAL Id: hal-00688919

https://hal.science/hal-00688919v2

Submitted on 30 Jul 2012

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Reduction algorithm for simplicial complexes

A. Vergne, L. Decreusefond and P. Martins
Institut Telecom, TELECOM ParisTech, LTCI
Paris, France
Email: avergne, decreuse, martins@telecom-paristech.fr

***Abstract*—In this paper, we aim at reducing power consumption in wireless sensor networks by turning off supernumerary sensors. Random simplicial complexes are tools from algebraic topology which provide an accurate and tractable representation of the topology of wireless sensor networks. Given a simplicial complex, we present an algorithm which reduces the number of its vertices, keeping its homology (i.e. connectivity, coverage) unchanged. We show that the algorithm reaches a Nash equilibrium, moreover we find both a lower and an upper bounds for the number of vertices removed, the complexity of the algorithm, and the maximal order of the resulting complex for the coverage problem. We also give some simulation results for classical cases, especially coverage complexes simulating wireless sensor networks.**

I. Introduction

Wireless sensor networks attract more and more research attention due to the extent of their applications as well as the decreasing costs and sizes of the electronic circuits. Fields where wireless sensor networks can be used range from battlefield surveillance to target enumeration in agriculture and include environmental monitoring. In most applications, the topology of the network, such as its connectivity and its coverage, is a critical factor. Moreover, sensors are autonomous systems: they are not plugged in nor physically connected to each other. Battery life is thus a key problem and energy saving a crucial point in wireless sensor networks management. Sensors are often deployed in large numbers, exceeding the number of necessary sensors. A first approach to reduce energy consumption would logically be to turn off sensors randomly. However by doing so one could modify the topology of the sensor network by creating a coverage hole or breaking the connectivity. Therefore, we first have to know the network's topology before we can figure out energy saving methods. To guarantee the full knowledge of the topology one solution is to deploy the sensors according to a regular pattern (hexagon, square grid, rhombus or equilateral triangle) [1]. However the target field does not always allow such a precise deployment. Furthermore, the topology may not be time-invariant: sensors could be destroyed, their batteries could die, or their communication could be disturbed by seasonal changes. Another approach is then to consider a random deployment that may create clusters of sensors or on the contrary, may leave holes of coverage. There is thus extensive research on the coverage problem in wireless sensor networks. Location-based [9] and ranged-based [18] methods require exact location information for the former or exact distance between sensors for the latter. Connectivity-based schemes seem of greater interest since they do not require such knowledge. In [10], Ghrist et al. introduced the so-called Vietoris-Rips complex, which is only based on the proximity graph between sensors, and determined the coverage by computing the homology of this complex. Coverage computation via homology boils down to simple linear algebra computations. It is used in [5], [15] and [19] as a tool for a network operator to evaluate the quality of its network. A distributed version of some of these algorithms is presented in [17], [16] in order to detect coverage holes. From a mathematical point of view, moments of various variables, such as the Betti numbers or the Euler characteristic of the covered region, can be obtained in specific regimes [14], or explicitly in one dimension [6] and in any dimension by means of Malliavin calculus [7].

In this paper, we present an algorithm which returns which sensors can be turned off without modifying the topology of the network. Given a simplicial complex, our algorithm suppresses vertices in an optimized order, keeping the complex homology unchanged. We can see an example of execution of the algorithm in Figure 1.

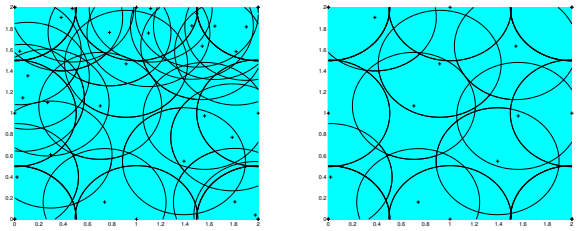


Fig. 1. A sensor network before and after the coverage reduction algorithm.

We show that the algorithm reaches a Nash equilibrium: Every vertex in the final abstract simplicial complex is needed to maintain the homology. We evaluate a lower and an upper bound for the number of removed vertices. The average complexity of the algorithm is analyzed for two kind of random simplicial complexes: Erdös-Rényi complexes and Poisson random geometric complexes. We show that this complexity is polynomial for the former and (slighlty) exponential for the latter. We also give the maximal order of the resulting complex for the coverage problem.

This is the first reduction algorithm based on simplicial

complexes using homology aimed at energy savings in wireless sensor networks. A classical approach to power management in networks is the usage of the connectivity graph, such as in the dominating graphs problem [12]. However, graphs are 2-dimensional objects. One vertex has full knowledge of its neighbors, but there is no representation of the interaction between these neighbors. Therefore there is no notion of coverage in graphs. Simplicial complexes allow us to represent higher order relations, and are thus more convenient for the representation of wireless sensor networks. Several works may seem at first glance related to our approach but they do not exactly fit with our purposes. In [8], [13], the authors use reduction of chain complexes to compute homology, reducing the work domain, which make it inapplicable to a coverage problem. Witness complexes reduction, which is a reduction to a chosen number of points, is used in [4] to compute topological invariant. In this work, as in the work on reduction of chains complexes, authors use reduction to compute the homology, whereas we use homology results to reduce optimally a simplicial complex. Finally, the authors of [3] present a game theoretic approach to power management where they define a coverage function. However this method requires precise location information as well as coverage knowledge. Moreover, authors aim at identifying sub-optimal solutions that does not guarantee an unmodified coverage.

The paper is organized as follows: in Section 2, we introduce simplicial homology concepts and variables that we will use in the next sections. Section 3 is devoted to the description of our reduction algorithm. In Section 4, we discuss its mathematical properties. Some simulation results are given in Section 5.

II. Simplicial homology

Graphs can be generalized to more generic combinatorial objects known as simplicial complexes. While graphs model binary relations, simplicial complexes represent higher order relations. A simplicial complex is a combinatorial object made up of vertices, edges, triangles, tetrahedra, and their n-dimensional counterparts. Given a set of vertices V and an integer k, a k-simplex is an unordered subset of $k+1$ vertices $[v_0, v_1 \ldots, v_k]$ where $v_i \in V$ and $v_i \neq v_j$ for all $i \neq j$. Thus, a 0-simplex is a vertex, a 1-simplex an edge, a 2-simplex a triangle, a 3-simplex a tetrahedron, etc.

Any subset of vertices included in the set of the $k+1$ vertices of a k-simplex is a face of this k-simplex. Thus, a k-simplex has exactly $k+1$ $(k-1)$- faces, which are $(k-1)$-simplices. For example, a tetrahedron has four 3-faces which are triangles. A simplicial complex is a collection of simplices which is closed with respect to the inclusion of faces, i.e. all faces of a simplex are in the set of simplices, and whenever two simplices intersect, they do so on a common face. An abstract simplicial complex is a purely combinatorial description of the geometric simplicial complex and therefore does not need the property of intersection of faces. For details about algebraic topology, we refer to [11].

One can define an orientation for an abstract simplicial complex, where a change in the orientation corresponds to a change in the sign of the coefficient:

$$[v_0, \ldots, v_i, \ldots, v_j, \ldots, v_k] = -[v_0, \ldots, v_j, \ldots, v_i, \ldots, v_k].$$

Definition 1: Given an abstract simplicial complex X, for each integer k, $C_k(X)$ is the vector space spanned by the set of oriented k-simplices of X.

Definition 2: The boundary map ∂_k is defined to be the linear transformation $\partial_k : C_k \to C_{k-1}$ which acts on basis elements $[v_0, \ldots, v_k]$ of C_k via

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

This map gives rise to a chain complex: a sequence of vector spaces and linear transformations

$$\ldots \xrightarrow{\partial_{k+2}} C_{k+1} \xrightarrow{\partial_{k+1}} C_k \xrightarrow{\partial_k} C_{k-1} \xrightarrow{\partial_{k-1}} \ldots \xrightarrow{\partial_1} C_0 \xrightarrow{\partial_0} 0.$$

Definition 3: The k-th boundary group of X is $B_k(X) = \operatorname{im} \partial_{k+1}$.

Definition 4: The k-th cycle group of X is $Z_k(X) = \ker \partial_k$.

A standard result asserts that for any integer k,

$$\partial_k \circ \partial_{k+1} = 0.$$

It follows that $B_k \subset Z_k$.

Definition 5: The k-th homology group of X is the quotient vector space:

$$H_k(X) = \frac{Z_k(X)}{B_k(X)}.$$

Definition 6: The k-th Betti number of X is the dimension:

$$\beta_k = \dim H_k = \dim Z_k - \dim B_k.$$

We can compute the Betti numbers in a simple case. Let X be the simplicial made up of 5 vertices $[v_0], \ldots, [v_4]$, 6 edges $[v_0, v_1]$, $[v_0, v_2]$, $[v_1, v_2]$, $[v_1, v_4]$, $[v_2, v_3]$ and $[v_3, v_4]$, and one triangle $[v_0, v_1, v_2]$.

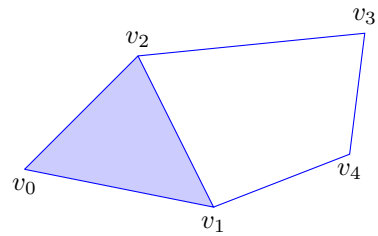


Fig. 2. A geometric representation of X

The boundary maps associated to the simplicial complex X are in matrix formulation:

$$\partial_1 = \begin{array}{c} \\ [v_0] \\ [v_1] \\ [v_2] \\ [v_3] \\ [v_4] \end{array} \begin{array}{c} \begin{array}{cccccc} [v_0v_1] & [v_0v_2] & [v_1v_2] & [v_1v_4] & [v_2v_3] & [v_3v_4] \end{array} \\ \begin{pmatrix} -1 & -1 & 0 & 0 & 0 & 0 \\ 1 & 0 & -1 & -1 & 0 & 0 \\ 0 & 1 & 1 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 1 & -1 \\ 0 & 0 & 0 & 1 & 0 & 1 \end{pmatrix} \end{array},$$

$$\partial_2 = \begin{array}{c} \\ [v_0, v_1] \\ [v_0, v_2] \\ [v_1, v_2] \\ [v_1, v_4] \\ [v_2, v_3] \\ [v_3, v_4] \end{array} \begin{array}{c} [v_0, v_1, v_2] \\ \begin{pmatrix} 1 \\ -1 \\ 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} \end{array}.$$

The boundary map ∂_0 is the null function on the set of vertices. Then we can compute:

$$\begin{aligned} \beta_0(X) &= \dim\ker\partial_0 - \dim\operatorname{im}\partial_1 \\ &= 5 - 4 \\ &= 1 \\ \beta_1(X) &= \dim\ker\partial_1 - \dim\operatorname{im}\partial_2 \\ &= 2 - 1 \\ &= 1 \end{aligned}$$

There are several famous types of abstract simplicial complexes, here we focus on two particular abstract simplicial complexes defined on a metric space.

Definition 7 (Čech complex): Given (X, d) a metric space, ω a finite set of points in X, and ϵ a real positive number. The Čech complex of parameter ϵ of ω, denoted $\mathcal{C}_\epsilon(\omega)$, is the abstract simplicial complex whose k-simplices correspond to $(k+1)$-tuples of vertices in ω for which the intersection of the $k+1$ balls of radius ϵ centered at the $k+1$ vertices is non-empty.
Thus the Čech complex characterizes the coverage of a domain as we can see in Figure 3.

Definition 8 (Vietoris-Rips complex): Given (X, d) a metric space, ω a finite set of points in X, and ϵ a real positive number. The Vietoris-Rips complex of parameter ϵ of ω, denoted $\mathcal{R}_\epsilon(\omega)$, is the abstract simplicial complex whose k-simplices correspond to unordered $(k+1)$-tuples of vertices in ω which are pairwise within distance less than ϵ of each other.

In general, unlike the Čech one, Vietoris-Rips complexes are not topologically equivalent to the coverage of an area. However, the following gives us the relation between coverage and Vietoris-Rips complexes:

Lemma 1: Given (X, d) a metric space, ω a finite set of points in X, and ϵ a real positive number,

$$\mathcal{R}_{\sqrt{3}\epsilon}(\omega) \subset \mathcal{C}_\epsilon(\omega) \subset \mathcal{R}_{2\epsilon}(\omega).$$

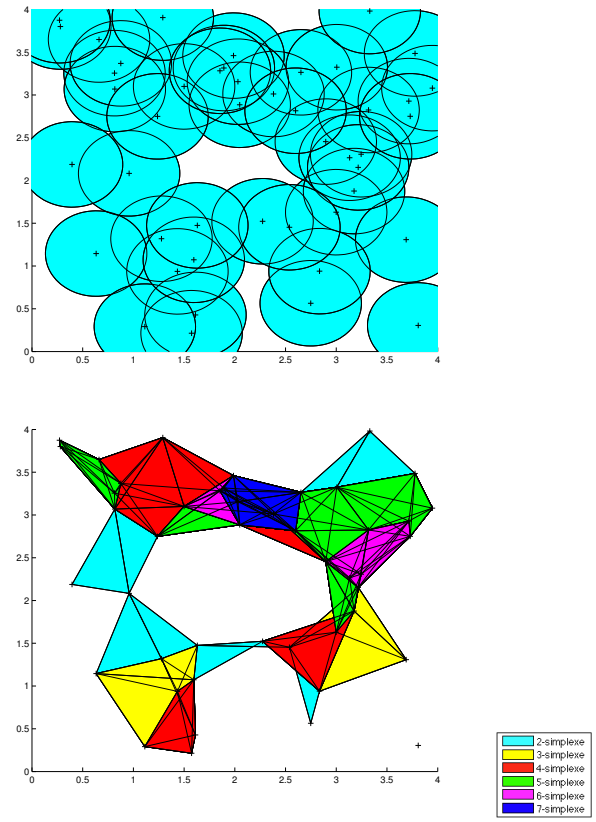


Fig. 3. A sensor network and its associated Čech complex.

Only graph description is required to build a Vietoris-Rips complex. In the same way we can build an abstract simplicial complex from any graph: each k-simplex is included in the complex if every one of its $(k-1)$-faces already are.

The Betti numbers are used to count the number of k-dimensional holes. For example, β_0 counts the number of 0-dimensional holes, that is the number of connected components. And β_1 counts the number of holes in the plane. If we are in dimension d, the k-th Betti number for $k \geq d$ has no geometric meaning.

III. Reduction algorithm

In this section, we present the reduction algorithm. The algorithm takes as input a fully described abstract simplicial complex: all k-simplices must be enumerated for every integer k. Then the algorithm aims at removing superfluous vertices while maintaining the homology type of the abstract simplicial complex.

There are several levels of knowledge of the homology type: you might want to maintain only the same first k_0-th homology group and Betti numbers. For the ease of presentation, we restrict ourselves to dimension 2, so we consider only the first two Betti numbers: β_0 is the number of connected components, β_1 is the number of holes in the plane. We have thus two different algorithms, for $k_0 = 1$ and $k_0 = 2$, but the main idea can be extended to greater dimensions.

The first algorithm, henceforth referred to as the connectivity algorithm, aims at maintaining β_0. The connectivity algorithm takes as inputs the abstract simplicial complex, plus the list of connected active vertices that are to stay connected.

The second algorithm, called the coverage algorithm, aims at maintaining β_0 and β_1. It takes as inputs in addition to the abstract simplicial complex, the list of connected vertices which define the boundary of the area that is to stay covered.

A. Degree calculation

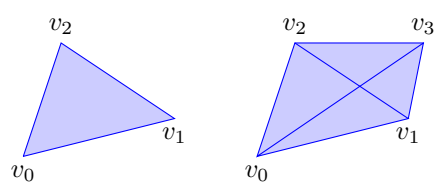
The first part of the algorithm is the calculation of a degree that we define for every k_0-simplex, with k_0 the number of Betti numbers to be kept unchanged:

Definition 9: For k integer, the degree of a k-simplex $[v_0, v_1, \ldots, v_k]$ is the size of the greatest simplex it is part of:

$$D[v_0, v_1, \ldots, v_k] = \max\{d \mid [v_0, v_1, \ldots, v_k] \subset d\text{-simplex}\}.$$

We immediately have $D[v_0, v_1, \ldots, v_k] \geq k$ for any k-simplex.

We can see examples of value of the degree for 2-simplices in figure 4.



$D[v_0, v_1, v_2] = 3$ $\quad$ $D[v_0, v_1, v_2] = 4$

Fig. 4. Example of values of degrees of triangles

For the remainder of the paper, let $s_k(X)$, or s_k, be the number of k-simplices of the abstract simplicial complex X.

Let us denote $D_1, \ldots, D_{s_{k_0}}$ the s_{k_0} degrees of a complex, their computation is done as shown in Algorithm 1.

Algorithm 1 Degree calculation

```
for i = 1 → s_{k_0} do
    Get (v_0, ..., v_{k_0}) the vertices of the i-th k_0-simplex
    k = k_0
    while (v_0, ..., v_{k_0}) are vertices of a (k + 1)-simplex do
        k = k + 1
    end while
    D_i = k
end for
return D_1, ..., D_{s_{k_0}}
```

B. Indices computation

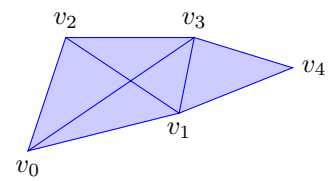
The second part of the algorithm is the computation of the indices that we define for each vertex:

Definition 10: The index of a vertex v is the minimum of the degrees of the k_0-simplices it is a vertex of:

$$I[v] = \min\{D[v_0, v_1, \ldots, v_{k_0}] \mid v \in [v_0, v_1, \ldots, v_{k_0}]\},$$

with k_0 the number of Betti numbers to be kept unchanged. If a vertex v is not a vertex of a k_0-simplex then $I[v] = 0$.

We can see in figure 5 an example of value of indices in an abstract simplicial complex.



$$D[v_0, v_1, v_2] = D[v_0, v_1, v_3] = D[v_0, v_2, v_3] = D[v_1, v_2, v_3] = 4$$
$$D([v_1, v_3, v_4]) = 3$$
$$I[v_0] = I[v_2] = 4 \text{ and } I[v_1] = I[v_3] = I[v_4] = 3$$

Fig. 5. Example of values of indices

Let $v_1, v_2, \ldots, v_{s_0}$ be the vertices of the abstract simplicial complex, the computation of the s_0 indices is done as shown in Algorithm 2.

Algorithm 2 Indices computation

```
for i = 1 → s_0 do
    I[v_i] = 0
    for j = 1 → s_{k_0} do
        if v_i is vertex of k_0-simplex j then
            if I[v_i] == 0 then
                I(v_i) = D_j
            else
                I[v_i] = min{I[v_i], D_j}
            end if
        end if
    end for
end for
return I[v_1], ..., I[v_{s_0}]
```

The index of a vertex is an indicator of the density of vertices around the vertex: an index of k_0 indicates that at least one k_0-simplex of the vertex is not a face of any $(k_0 + 1)$-simplices; whereas a high index shows that each k_0-simplices of the vertex is part of higher simplices.

An index of zero indicates that the vertex is not a part of a any k_0-simplex: the vertex is isolated up to the k_0-th degree. For $k_0 = 1$, that means that the vertex is disconnected. For

$k_0 = 2$, the vertex is only linked to other vertices by edges, therefore is inside a coverage hole.

C. Optimized order for the removal of vertices

We now consider the entire algorithm and look for an optimized order to remove vertices which are superficial to the homology.

First, we consider the particular vertices. The active or boundary vertices that are given as inputs and define the area of interest for the homology computation are critical vertices that are requested to remain in the last complex. They are flagged as irremovable by a negative index.

Then the indices give us an order for the removal of vertices: the greater the index of a vertex, the more likely it is superfluous for the homology of the abstract simplicial complex. So the vertices with the greatest index are candidates for removal: one is chosen randomly. The removal of a vertex leads to the degradation of all the k-simplices it was a vertex of, to $(k-1)$-simplices for every integer k.

We need to ensure that the homology is unchanged, so we compute the k_0 first Betti numbers thanks to the boundary maps once before the algorithm runs, then every time a vertex is removed. This computation is instantaneous since the complex is already built, and only adjacency matrices defining the complex are needed. If the homology changes, the vertex is put back into the abstract simplicial complex, with a negative index to flag it as critical. Otherwise, the removal of the vertex is confirmed. The degrees of the k_0-simplices and the indices of the vertices are recalculated for the new abstract simplicial complex.

Lemma 2: When a vertex of index $I > k_0$ is removed, only the vertices sharing an I-simplex with it, and of index I can have their index changed to $I-1$.

Proof: Let w be the removed vertex of index I, and v any vertex of the abstract simplicial complex. Since w is removed, I is the maximum index of the complex.

If v does not share any simplex with w, none of the degrees of its k_0-simplex will change, and neither will its index.

Thus let us consider that the maximum common simplex of v and w is a k-simplex. If $k < I$ then w have index $k < I$, which is absurd. Then we can assume that $k \geq I$. Either the index of v is strictly less than I and thus comes from a simplex not shared with w, therefore it is unmodified by the removal of w. Or the index of v is I, it either still comes from a I-simplex not shared with w and remains unmodified, or it comes from a common I-simplex. In the latter, after the removal of w, the index of v becomes $I-1$. ■

The algorithm goes on removing vertices until every remaining vertex has its index below or equal to $k_0 - 1$. We give in Algorithm 3 the whole algorithm for the conservation of the k_0 first Betti numbers.

Definition 11 (Dominating set): As defined in [12], a set $S \subseteq V$ of vertices of a graph $G = (V, E)$ is called a dominating set if every vertex $v \in V$ is either an element of S or is adjacent to an element of S.

Algorithm 3 Reduction algorithm

input: Abstract simplicial complex X, list L_C of critical vertices (active or boundary).
  Computation of $\beta_0(X), \beta_1(X), \ldots, \beta_{k_0-1}(X)$
  Computation of $D_1(X), \ldots, D_{s_{k_0}}(X)$
  Computation of $I[v_1(X)], \ldots, I[v_{s_0}(X)]$
  for all $v \in L_C$ **do**
    $I[v] = -1$
  end for
  $I_{\max} = \max\{I[v_1(X)], \ldots, I[v_{s_0}(X)]\}$
  while $I_{\max} > k_0 - 1$ **do**
    Draw w a vertex of index $I_{\max}$
    $X' = X \backslash \{w\}$
    Computation of $\beta_0(X'), \beta_1(X'), \ldots, \beta_{k_0-1}(X')$
    if $\beta_i(X') \neq \beta_i(X)$ for any $i = 0, \ldots, k_0 - 1$ **then**
      $I[w] = -1$
    else
      Computation of $D_1(X'), \ldots, D_{s'_{k_0}}(X')$
      for $i \to 1 : s'_0$ **do**
        if $I[v_i(X')] \geq 0$ **then**
          Computation of $I[v_i(X')]$
        end if
      end for
      $I_{\max} = \max\{I[v_1(X')], \ldots, I[v_{s'_0}(X')]\}$
      $X = X'$
    end if
  end while
  return X

Definition 12 (Full domain hypothesis): Let us call the full domain hypothesis when the critical vertices define the whole input space: for the connectivity algorithm, that means that the active vertices are a dominating set of the initial abstract simplicial complex. For the coverage algorithm, that means that all the vertices are in the convex hull defined by the boundary vertices.

Lemma 3: Under the full domain hypothesis, the algorithm can stop when all vertex indices are below or equal to k_0 instead of $k_0 - 1$.

Proof: Let us suppose the input data satisfies the full domain hypothesis. Let v be a vertex of index k_0, which means that at least one of the k_0-simplex it is vertex of is not a face of any (k_0+1)-simplex. The removal of this vertex would lead to the removal of this particular k_0-simplex. Since we need to maintain the homology on the entire domain, this would lead to a k_0-dimensional hole, and an increment of β_{k_0-1}. ■

IV. Properties

Theorem 4 (Nash equilibrium): Our algorithm reaches a Nash equilibrium as defined in [3]: every vertex in the final abstract simplicial complex is needed to maintain its homology.

Proof: In the final abstract simplicial complex, every vertex is of index less than or equal to $k_0 - 1$. By the definition of a computed index, it is impossible for an index to be strictly

less than k_0 if nonzero or negative. We then differentiate two types of vertices: vertices of index -1 and 0.

First, negative indices are given to vertices to flag them as critical. Either a vertex is of negative index because it is an active/boundary vertex, in which case it is required to stay in the complex. Or a vertex is of negative index if its removal leads to a change in the Betti numbers.

Then, a vertex of index 0 is an isolated vertex. If it isolated up to the k_0-th of degree, its removal will decrease the k_0th Betti number. For example, the removal of a first degree isolated vertex, that is a disconnected one, would lead to the decrement of β_0. As well, the removal of a vertex inside a hole will lead to the union of 2 or more holes.

In the case of lemma 3, its proof shows that the vertices of indices k_0 are needed to maintain the homology. ■

Theorem 5 (Upper and lower bounds): Let E_k be the set of vertices that have indices k. The number of removed vertices M is bounded by:

$$\sum_{k=k_0}^{I_{\max}} \mathbf{1}_{[E_k \neq \emptyset]} \leq M \leq \sum_{k=k_0}^{I_{\max}} |E_k|.$$

Proof: Let us begin with the upper bound, the algorithm runs until all indices are less than or equal to $k_0 - 1$. So the maximum number of vertices the algorithm can remove is the number of vertices that initially have their index strictly greater than $k_0 - 1$. This is an optimal upper bound since this number of removed vertices is reached in the following case:

Let a k-simplex, with $k > k_0$, be the initial abstract simplicial complex, and n_C of its vertices be the initial critical vertices, necessarily $n_C \leq k+1$. The n_C critical vertices have negative indices, the $k+1-n_C$ other vertices have an index of k, and they are all removed.

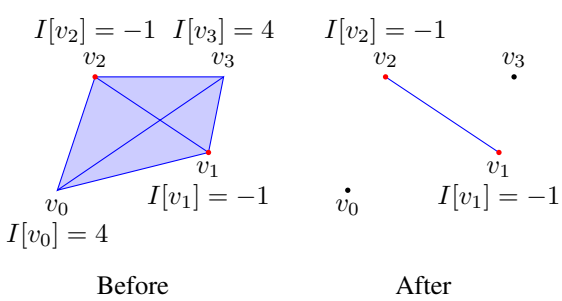


Fig. 6. Example of this case with $k = 3$ and $n_C = 2$, the 2 nodes v_0 and v_3 are removed by the algorithm.

For the lower bound, we have seen in Lemma 2 that the removal of a vertex of index $I_{\max}$ can only decrease the index of vertices of index $I_{\max}$. In the worst case, it decreases all indices $I_{\max}$ and the value of $I_{\max}$ changes, not necessarily to $I_{\max} - 1$ depending on the critical vertices. Thus we can see, that at least one vertex per index value can be removed, hence the result.

The lower bound is reached in the previous case if $n_C = k$. ■

Theorem 6 (Complexity): The complexity of the algorithm is upper bounded by:

$$(n - n_C) s_{k_0} \left(n + \sum_{k=k_0+1}^{H} s_k \right),$$

with n_C being the number of initial critical vertices, s_k the number of k-simplices in the abstract simplicial complex, $n = s_0$ the total number of vertices, and H the highest simplex in the complex.

Proof: For the computation of the degrees of every k_0-simplex, the algorithm traverses all the k-simplices for $k_0 < k \leq H$ to see if the k_0-simplex is included in it. Since there is s_k k-simplices, the computation of the degrees is of complexity $s_{k_0} \sum_{k=k_0+1}^{H} s_k$.

For the computation of the indices, the algorithm traverses, for every one of the n vertices, the k_0-simplices it is vertex of, which is at most all the k_0-simplices. The complexity of the computation of all the indices is therefore $s_{k_0} n$.

These computations are done before every removal of vertices, which is at most the total number of vertices minus the number of initial critical vertices, hence the result. ■

Corollary 7: The complexity of the algorithm is upper bounded by 2^n when n goes to infinity.

Proof: This is a direct consequence of Theorem 6 since the number of k-simplices s_k can be upper bounded by $\binom{n}{k+1}$. ■

Corollary 8: If we apply our algorithm to a Čech complex $\mathcal{C}_\epsilon(\omega)$, defined with the maximum norm, based on a Poisson point process of parameter λ on a torus of side a in dimension d. The complexity of the algorithm is of the order of $O((1 + (\frac{2\epsilon}{a})^d)^n)$ on average when n goes to infinity.

Proof: According to [7], we have:

$$\mathbf{E}_\lambda \left[s_{k-1} \,|\, s_0 = n \right] = \binom{n}{k} k^d \left(\frac{2\epsilon}{a} \right)^{d(k-1)},$$

$$\mathrm{Cov}_\lambda \left[s_{k-1}, s_{l-1} \,|\, s_0 = n \right] = \sum_{i=1}^{k \wedge l} \binom{n}{k+l-i} \binom{k+l-i}{k} \binom{k}{i}$$
$$\left(k + l - i + 2 \frac{(k-i)(l-i)}{i+1} \right)^d \left(\frac{2\epsilon}{a} \right)^{d(k+l-i-1)}.$$

Plugging this into the complexity formula of Theorem 6 gives the result. ■

We define the Erdös-Rényi complex based on the eponymous random graph:

Definition 13 (Erdös-Rényi complex): Given n an integer and p a real number in $[0, 1]$, the Erdös-Rényi complex of parameters n and p, denoted $G(n, p)$, is an abstract simplicial complex with n vertices which are connected randomly. Each edge is included in the complex with probability p independent from every other edge. Then a k-simplex, for $k \geq 2$, is included in the complex if and only if all its faces already are.

Corollary 9: The complexity of the algorithm for the Erdös-Rényi complex based on the graph $G(n,p)$ is of the order of $O(n^{2(k_0+2)})$ on average when n goes to infinity.

Proof: In the Erdös-Rényi complex, the expected values of the number of k-simplices and of the product of numbers of k-simplices are given in [2]:

$$\begin{aligned}\mathbf{E}_\lambda\left[s_{k-1}\right] &= \binom{n}{k}p^{\binom{k}{2}},\\ \mathbf{E}_\lambda\left[s_{k-1}s_{l-1}\right] &= \sum_{i=0}^{k\wedge l}\binom{n}{k}\binom{k}{i}\binom{n-k}{l-i}p^{\binom{k}{2}+\binom{l}{2}-\binom{i}{2}}.\end{aligned}$$

The results then comes from the complexity formula of Theorem 6. ■

Theorem 10 (Highest order simplex): In the case of the algorithm applied to a Vietoris-Rips complex or a Čech complex in dimension d, the highest order simplex of the final complex is at most a $(2d-1)$-simplex, excluding critical vertices.

Proof: Let k be an integer strictly greater than $2d-1$, and let us suppose that the $k+1$ non-critical vertices $v_0,\ldots,v_k$ form a k-simplex.

Since we are in dimension d, we can consider the vertex the furthest on each of the $2d$ directions. Note that two of these vertices can be the same vertex. Let us reorder the vertices such that $v_0,\ldots,v_{i-1}$ with $i\leq 2d$ are the extremity vertices defined above. Then $v_i,\ldots,v_k$ must fall in the covered area of $v_0,\ldots,v_{i-1}$. Such vertices can be removed without changing the homology of the complex, therefore they can not be in the final complex hence a contradiction. ■

Corollary 11: In the case of the algorithm applied to a Vietoris-Rips complex or a Čech complex under the full domain hypothesis in 2 dimensions, to make the final simplicial complex planar, it is necessary and sufficient to remove edges that are second diagonals of a square.

Proof: This is a direct consequence of Theorem 10. ■

Theorem 12: For the algorithm applied to a Vietoris-Rips complex or a Čech complex under the full domain hypothesis, the set of remaining vertices in the final complex is a dominating set of the set of vertices in the initial complex.

Proof: Let ϵ be the parameter of the Vietoris-Rips complex. The area inside the coverage boundary is covered by the final abstract simplicial complex: each point of this area is at distance lower than ϵ from at least three vertices of the final complex, znd is thus adjacent to them. This is true for every vertex of the initial complex.

The proof is the same in the case of the Čech complex. ■

V. Simulation results

Our simulations aim to illustrate our algorithm. The results are highly dependent on the chosen parameters: for the connectivity algorithm the percentage of removed vertices is linked to the probability that the active vertices are connected without any intermediary in the complex; for the coverage algorithm, it is linked to the ratio initial number of vertices on number of vertices needed for the coverage. We simulated our algorithm on two types of complexes: the connectivity algorithm on the Erdös-Rényi complex, and the coverage algorithm on the Vietoris-Rips complex.

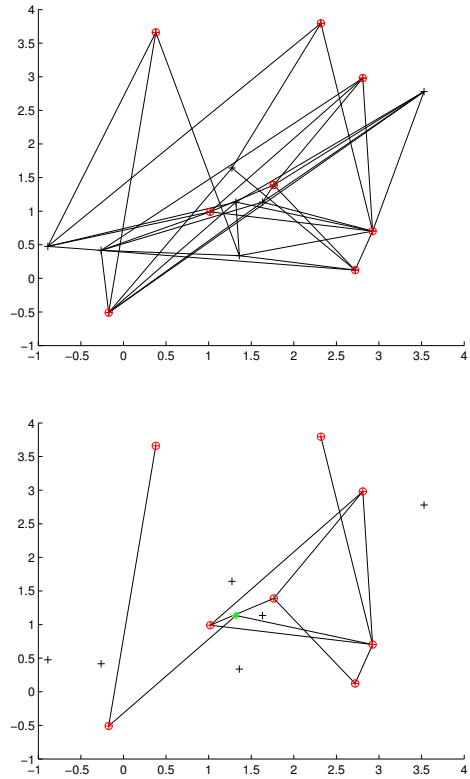


Fig. 7. A Erdös-Rényi complex before and after the connectivity algorithm.

We can see in Figure 7 one realization of the connectivity algorithm on a Erdös-Rényi complex of parameter $n=15$ and $p=0.3$, with a random set of active vertices. A vertex is active with probability $p_a=0.5$ independently from every other vertices. We chose a small number of vertices for the figure to be readable. Active vertices are circled, and non-active vertices which are kept to maintain the connectivity between active vertices are starred.

With the parameters $n=60$ vertices, and $p=0.2$, on average on 1000 configurations without disconnectivity with p_a taking values between 0.1 and 0.5 (200 configurations each), the algorithm removed 98% of the non-active vertices:

p_a	% of removed vertices
0.1	94.96%
0.2	97.14%
0.3	98.59%
0.4	99.43%
0.5	99.87%

The behaviour of the connectivity algorithm on a Vietoris-Rips complex is analogous.

For the coverage algorithm on a Vietoris-Rips complex, we simulated the set of vertices using a Poisson point process:

Definition 14: A Poisson point process ω with intensity λ on a Borel set X is defined by:

i) For any $A \in \mathcal{B}(X)$, the number of vertices in A, $\omega(A)$, is a random variable following a Poisson law of parameter $\lambda S(A)$,

$$\mathbf{P}(\omega(A) = k) = e^{\lambda S(A)} \frac{(\lambda S(A))^k}{k!}.$$

ii) For any disjoint A, $A' \in \mathcal{B}(X)$, the random variables $\omega(A)$ and $\omega(A')$ are independent.

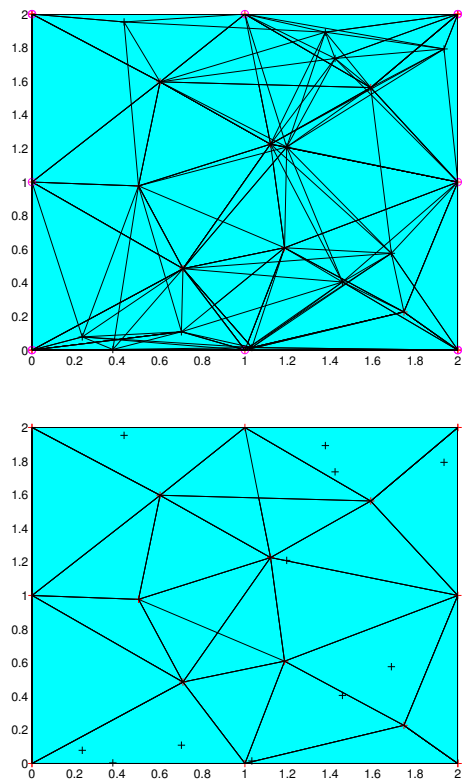


Fig. 8. A Vietoris-Rips complex before and after the coverage algorithm.

We can see in Figure 8 one realization of the coverage algorithm on a Vietoris-Rips complex of parameter $\epsilon = 1$ based on a Poisson point process of intensity $\lambda = 4.2$ on a square of side length $a = 2$, with a fixed boundary of vertices on the square perimeter. The boundary vertices are circled.

With the parameters, chosen to ensure coverage, $\epsilon = 1$, $\lambda = 5.1$ and $a = 2$, on average on 1000 configurations, the algorithm removed 69.35% of the non-boundary vertices.

References

[1] X. Bai, S. Kumar, D. Xuan, Z. Yun, and T. H. Lai. Deploying wireless sensors to achieve both coverage and connectivity. In *Proceedings of the 7th ACM international symposium on Mobile ad hoc networking and computing*, MobiHoc '06, pages 131–142, New York, NY, USA, 2006. ACM.

[2] B. Bollobás and P. Erdös. Cliques in random graphs. *Mathematical Proceedings of the Cambridge Philosophical Society*, 80(3):419–427, 1976.

[3] E. Campos-Nañez, A. Garcia, and C. Li. A game-theoretic approach to efficient power management in sensor networks. *Oper. Res.*, 56(3):552–561, 2008.

[4] V. de Silva and G. Carlsson. Topological estimation using witness complexes. *IEEE Symposium on Point-based Graphic*, pages 157–166, 2004.

[5] V. de Silva and R. Ghrist. Coordinate-free coverage in sensor networks with controlled boundaries via homology. *International Journal of Robotics Research*, 25, december 2006.

[6] L. Decreusefond and E. Ferraz. On the one dimensional Poisson random geometric graph. *Journal of Probability and Statistics*, 2011. Art. ID 350382, 21.

[7] L. Decreusefond, E. Ferraz, H. Randriambololona, and A. Vergne. Simplicial homology of random configurations. http://hal-institut-telecom.archives-ouvertes.fr/hal-00578955, May 2012.

[8] P. Dłotko, R. Ghrist, M. Juda, and M. Mrozek. Distributed computation of coverage in sensor networks by homological methods. *Applicable Algebra in Engineering, Communication and Computing*, pages 1–30. 10.1007/s00200-012-0167-7.

[9] Q. Fang, J. Gao, and L. Guibas. Locating and bypassing routing holes in sensor networks. In *INFOCOM 2004. Twenty-third AnnualJoint Conference of the IEEE Computer and Communications Societies*, volume 4, pages 2458 – 2468 vol.4, march 2004.

[10] R. Ghrist and A. Muhammad. Coverage and hole-detection in sensor networks via homology. In *Proceedings of the 4th international symposium on Information processing in sensor networks*, IPSN '05, Piscataway, NJ, USA, 2005. IEEE Press.

[11] A. Hatcher. *Algebraic Topology*. Cambridge University Press, 2002.

[12] T. W. Haynes, S. T. Hedetniemi, and P. J. Slater. *Fundamentals of domination in graphs*, volume 208 of *Monographs and Textbooks in Pure and Applied Mathematics*. Marcel Dekker Inc., New York, 1998.

[13] T. Kaczyński, M. Mrozek, and M. Ślusarek. Homology computation by reduction of chain complexes. *Comput. Math. Appl.*, 35(4):59–70, 1998.

[14] M. Kahle. Random geometric complexes. *Discrete & Computational Geometry*, 45:553–573, 2011. 10.1007/s00454-010-9319-3.

[15] A. Muhammad and A. Jadbabaie. Decentralized computation of homology groups in networks by gossip. In *American Control Conference, 2007. ACC '07*, pages 3438 –3443, july 2007.

[16] F. Yan, P. Martins, and L. Decreusefond. Connectivity-based distributed coverage hole detection in wireless sensor networks. In *Global Telecommunications Conference (GLOBECOM 2011), 2011 IEEE*, pages 1 –6, dec. 2011.

[17] F. Yan, P. Martins, and L. Decreusefond. Accuracy of homology based approaches for coverage hole detection in wireless sensor networks. In *ICC 2012*, June 2012.

[18] C. Zhang, Y. Zhang, and Y. Fang. Detecting coverage boundary nodes in wireless sensor networks. In *Networking, Sensing and Control, 2006. ICNSC '06. Proceedings of the 2006 IEEE International Conference on*, pages 868 –873, 0-0 2006.

[19] A. Zomorodian and G. Carlsson. Computing persistent homology. *Discrete & Computational Geometry*, 33:249–274, 2005. 10.1007/s00454-004-1146-y.