Topological Data Processing for Distributed Sensor Networks with Morse-Smale Decomposition

Xianjin Zhu Rik Sarkar Jie Gao

Department of Computer Science, Stony Brook University. {xjzhu, rik, jgao}@cs.sunysb.edu

***Abstract*—We are interested in topological analysis and processing of the large-scale distributed data generated by sensor networks. Naturally, a large-scale sensor network is deployed in a geometric region with possibly holes and complex shape, and is used to sample some smooth physical signal field. We are interested in both the topology of the discrete sensor field in terms of the sensing holes (voids without sufficient sensors deployed), as well as the topology of the signal field in terms of its critical points (local maxima, minima and saddles). Towards this end, we develop distributed algorithms to construct the Morse-Smale decomposition, and study the performance benefits obtained by this approach. The sensor field is decomposed into *simply-connected* pieces, inside each of which the sensor signal is *homogeneous*, i.e., the data flows uniformly from a local maximum to a local minimum. The Morse-Smale decomposition can be efficiently constructed in the network locally, after which applications such as iso-contour queries, data-guided navigation and routing, data aggregation, and topologically faithful signal reconstructions benefit tremendously from it.**

I. Introduction

In this paper we are interested in the topological features of spatially distributed sensor data. In many application settings such as environmental monitoring, the sensor readings can be regarded as a dense sampling of an underlying physical signal field that often exhibits strong spatial and temporal correlations. Such spatial characteristics are important for many applications of sensor networks, as they correspond to physically significant phenomena. For example, peaks indicate heat sources in a heap map or traffic jams in car density map. Users of sensor networks are often interested in data-related queries such as

- *Iso-contour query*: from a query node q, find the iso-contours at value x, or count/report iso-contour components at a given value/range. This can be used to discover for example high pollution areas for rescue workers, or group targets for police officers.
- *Data-guided navigation and routing*: find a path from a source node s to a destination node t with all values on the path within a user-specified range. This can be used for navigation of packets in the network (e.g., avoiding sensor nodes with low energy level), or navigation of users/vehicles in the physical environment (e.g., avoiding traffic jam).

The spatial features such as local maxima, minima or saddles and their relationship capture the topological complexities of the signal field. Abstractions of these spatial features will enable data-related queries mentioned above, and also allow aggressive data compression and reconstruction while still preserving the important topological characteristics, which is useful for efficient data delivery and storage.

In this paper we deal with the topological structures of the signal field (in terms of critical points) and the topological structures of the sensor field (in terms of holes) simultaneously. We apply Morse theory [4] in sensor network setting and use a communication-efficient distributed algorithm to decompose a sensor network to cells. Each cell is simply connected (i.e., has no holes) and homogeneous (i.e., the data flows uniformly from a local maximum to a local minimum). The cell adjacency information is captured and represented by the Morse-Smale complex, which is a compact structure with size proportional to the number of critical points in the signal field and the number of holes in the network. One can thus afford to disseminate the Morse-Smale complex to all sensor nodes in the network as a high-level summary of the signal topology. The homogeneous flow inside each cell gives a natural coordinate system with one set of coordinates along the greatest descent vector and the other set of coordinates along the isolines, the two of which interweave nicely as a Cartesian coordinate system in the cell, and are smoothly glued along the boundary with adjacent cells. Thus the coordinate system supports local and easy navigation or routing operations both inside and across the cells, that can be exploited by the iso-contours queries and the data-guided navigation. Together with the compact Morse-Smale complex available at each node, we immediately have a 2-level routing structure, akin to the virtual coordinate system built in the GLIDER algorithm for efficient point-to-point routing [2], such that a global routing decision (e.g., a value-restricted routing request) can first consult with the high-level structure to identify the cells to visit, with the actual routing implemented with this global guidance by local greedy routing scheme inside each cell. This achieves a nice balance in supporting functions requiring global information through local operations under energy conservation requirement.

Morse-Smale Decomposition theory. Morse theory [7], [8] deals with the relation between the topology of a smooth manifold and the critical points of a smooth real-valued function f defined on the manifold. A point p is a critical point if the tangent vector at p is zero. Following the gradient vectors, a *stable manifold* of a critical point a is defined as the union of a and all points flowing into a. Similarly, an *unstable manifold* of a critical point a is the union of a and all points flowing out of a. A Morse function f is called Morse-

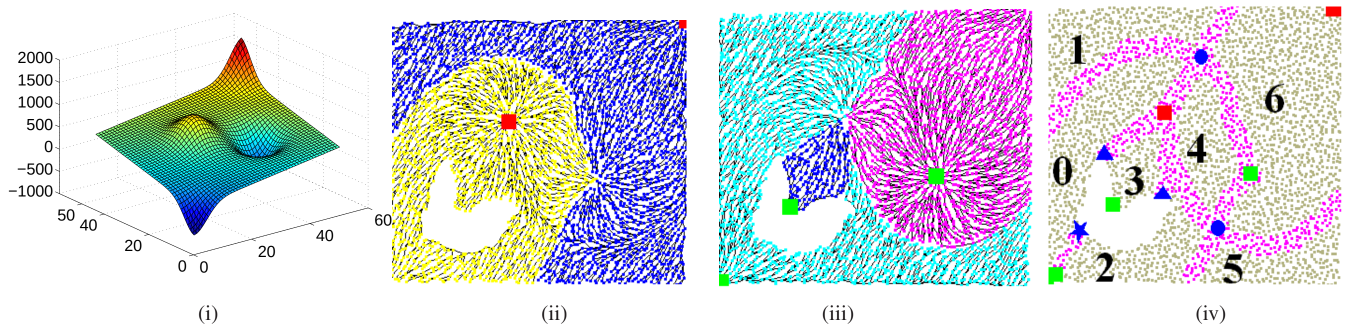


Fig. 1. (i) Original signal field. The network has a hole. (ii) Stable manifolds. (iii) Unstable manifolds. (iv) Morse-smale decomposition with each cell labeled. Red square: *max*; green square: *min*; blue disk: *regular saddle*; blue star: *max-saddle*; blue triangle: *min-saddle*.

Smale if the stable and unstable manifolds intersect only transversally. In this case, the Morse-Smale decomposition is the intersection of the stable and unstable manifolds. Each cell in the decomposition is a quadrangle with a local maximum, two saddles and a local minimum. This means all the gradient vectors in a cell are *uniform* – they all originate from the same maximum and flow into the same minimum. The Morse-Smale complex takes the dual of the decomposition of Morse-Smale cells and captures the topology of $\mathcal{M}$ through the study of the gradient of f.

When we apply Morse theory to a sensor field with holes, the good properties mentioned above do not directly carry over. The network holes can disrupt the Morse-Smale decomposition (for f defined in $\mathbb{R}^2$) in the sense that a cell may contain one or multiple holes in its interior and is no longer simply connected — thus causing problems with the Cartesian coordinate system as greedy routing can get stuck at the hole boundary and no longer deliver the message successfully. More critical points might be introduced by the holes as flows may end or start from hole boundaries.

In a companion paper [3], we developed the theory for Morse-Smale decomposition in a 2D region with boundaries that restores the good properties of such decomposition. Of special interest to the sensor network application is that we establish the connection of the Morse-Smale decomposition boundary with the 'cut locus' [1], [9] of the sensor data flow. Thus we identify a pair of *cut nodes* as two neighboring sensors with two different flows, either arriving at two different maxima/minima, or at the same maximum/minimum with different homotopy types (i.e., bypassing network holes in different ways). The cut pairs leading to different maxima/minima represent the boundaries of the stable manifolds of these different maxima/minima. The cut pairs leading to the same maximum/minimum but through flows of different homotopy types will further cut the holes open to make each cell homogeneous and simply connected. The identification of these cut nodes can be done with simply a sweep from all the maxima/minima. The cut pairs leading to different maxima/minima can be discovered by a simple check on the destinations of the flows through them. For the cut pairs with flows to the same critical point surrounding holes, we use similar ideas as in [9] to spot them. When the cut nodes are removed, the sensor field is left with multiple pieces, each piece has no hole and bears a homogeneous data flow. The Mrose-Smale complex captures how these pieces are glued to each other and represents the topology of the sensor field.

The algorithm for constructing the Morse-Smale decomposition and its dual for a sensor network does not require any information of where the holes are and how many of them. It is also more robust and cost-efficient than our previous approach [5] in handling the signal field topology and data-guided queries through the extraction of the contour tree (a tree on all the critical points of the signal field and captures how the connected components of the iso-contours merge/split as we increase/decrease the isovalue), as network holes may lead to asynchronous progress of the contour sweeps.

Contribution in this paper. Our companion paper [3] mainly laid out the rigorous definition for Morse-Smale decomposition for a 2D continuous region with holes and developed a distributed algorithm for a sensor field. This provides the theoretical foundation. In this paper we discuss the application of the Morse-Smale decomposition to sensor network problems:

- *Iso-contour queries and data-guided routing.* As explained earlier, the homogeneous flow inside each cell makes these routing primitives very easy with only local greedy decisions.
- *Sweeps for data collection and aggregation.* Sweeps can be used as a basic data collection and aggregation scheme in a sensor network [6]. With the Morse-Smale decomposition we can perform a sweep inside each cell that is simply connected, which guarantees the connectivity of the sweep frontier, thus reducing a lot of sensor active time on sweep frontier coordination and synchronization.
- *Topologically faithful compression and reconstruction.* If one records the values and positions of the nodes in the Morse-Smale decomposition boundary, it is easy to perform linear interpolation to reconstruct a signal field with precisely the same topological features. The reconstruction is not meant to be geometrically close but nevertheless can provide all the information that concerns the signal and the network topology.

We describe in detail how to make use of the Morse-Smale decomposition in these applications and provide performance improvement results in simulations.

II. Applications

In this section we describe the benefits of having a Morse-Smale decomposition and dual complex in different applications in a sensor network. In particular, we describe how the decomposition facilitates data-centric routing, aggregation by sweeps and signal reconstruction by decomposing the network into simple monotone pieces. Throughout the discussion, we assume a sensor field in a 2D domain with possibly network holes. Each node has a sensing value, which is a sample of a continuous signal field f at the sensor's location. The Morse-Smale decomposition partitions the sensor network into cells. Each cell is given an ID and all the nodes inside the cell know this ID. The dual complex specifies the adjacency information of these cells. In particular, if any two cells have some shared boundary, there are k edges connecting these two cells, with k equal to the number of connected components of the boundary. Recursively we can also establish the high-order simplices (such as triangles). For each edge in the dual complex, we associate a range value, which is the range of the nodes on the corresponding boundary component. We assume that the dual complex has been computed as mentioned in [3] and and is known at every node.

A. Data centric routing

We are interested in two types of data dependent queries :

- **Value restricted routing.** Find a path from a source node s to a destination node t with all values on the path within a user-specified range.
- **Iso-contour query.** From a query node q, find the iso-contours at value v, or count or report iso-contour components at the given value or range. This involves local determination of contour components and routing to each of these components.

To find a route in the sensor network, we first look at a sequence of cells to visit in the Morse-Smale complex, called a dual path. For better load balancing, we take a random dual path and avoid neighboring cell pairs whose intersection is a single point, and take only those neighbors that share an edge. With the help of the Morse-Smale decomposition, the following routing primitives can be easily implemented.

Definition 2.1. ***Routing primitives***

- ***Nodes on a contour** v.* *A node p is said to be on a contour v if the value at p is v, or if the value of p is greater than v, and p has some neighbor whose value is less than v.*
  This definition establishes the contours in terms of nodes in the network and will be used in routing and detecting iso-contours.
- ***Routing to an iso-contour** v.* *Given a node p in a cell c_p, routing to a contour within the cell consists of routing to a random neighbor of p with higher value iteratively, until a node on the contour v is reached.*
- ***Routing to a neighboring cell.*** *This is executed in two steps. First, from the union of ranges of edges shared with the neighboring cell, select a random contour value, and route to a node on that contour. Next, follow nodes on the contour to reach the destination neighbor cell. This is possible by the properties of Morse-Smale decomposition.*
- ***Routing to a node** q **in the same cell.*** *This is achieved by first reaching an iso-contour at the value of q, and then following the iso-contour to reach q.*

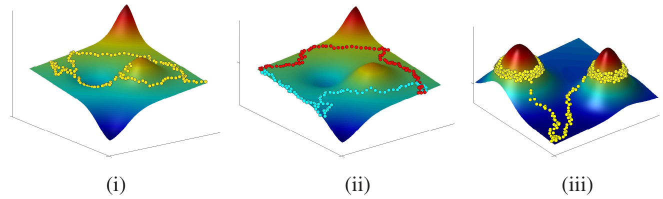


Fig. 2. (i) Two very different routes are utilized on two different routing requests. This result of randomization helps in load balancing. (ii) Depending on the range restriction applied, the algorithm constructs different paths. The red path is in response to a request for a path in a *high* range, the blue path on request for a *low* path. (iii) Iso-contours and routes to the different iso-contours found by the algorithm.

With these definitions in place, it is now possible to route from any node to an arbitrary cell or node in the network, by first computing a random dual path, and then traversing the dual path by means of the low-level routing primitives described above.

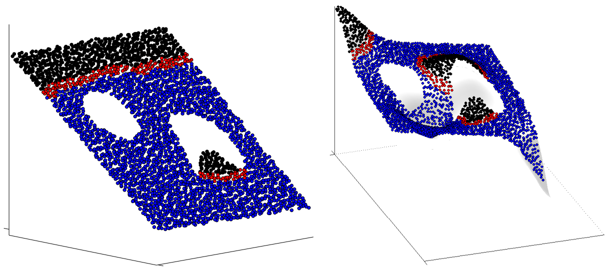
1) Value restricted routing: This is done simply by suitable restrictions on the routing primitives described above. Given a routing request, with values restricted to $[a, b]$ we perform all the routing steps relative to this range. Figure 2 (ii) shows a case where different range restrictions result in different paths.

2) Iso-contour queries: Iso-contour queries can be executed in two steps. First, identify the cells that contain some portion of the iso-contour in question, this can be done simply observing the maximum and minimum values in each cell. Now, several such cells may contain parts of a single connected contour component. We can identify the disjoint components using the following principle : *two of the already identified cells are neighbors and they share an edge whose range contains the queried level, iff the corresponding iso-contours belong to the same connected component.* This fact can be used to identify the connected components by a simple DFS in time linear in the size of the dual graph. Once the connected components of iso-contours have been identified, we can next select one cell of each component and route to it and to the contour level by the methods above. Figure 2 (iii) shows an example.

In this subsection we have presented examples on simple networks without *holes* for ease of understanding and graphic representation. The routing mechanisms however, are independent of such issues, and work equally well in presence of holes.

B. Data aggregation by sweeps

The idea of using a sweep over a sensor network to perform data aggregation was proposed in [6], where this method was shown to be more efficient than standard aggregation tree based methods. Please see [6] for the details. The intuition behind performing a sweep is to schedule the transmissions in a way that reduces collisions and energy usage. The time that a node must stay *active* is the duration that it lies on the



(i) (ii)

Fig. 3. Sweep in progress under different environments. *Black: swept nodes; blue:nodes not swept yet; red: active nodes*. (i) Shows a sweep by a geographic coordinate, where one sweep frontier has to wait at a saddle for another frontier to arrive. (ii) shows a similar case, but in sweep according to a signal value.

sweep frontier. Minimizing this up-time can greatly reduce power consumption and improve network life time.

The function f which determines the sweep can be an arbitrary one, with the understanding that the sweep will end at the minima of this function. It can simply be one of the geographic coordinates, producing a uniform slope (Fig. 3(i)) or a harmonic potential (see [6] for details) or simply an existing signal field (Fig. 3(ii)). The advantage of this last method is that it requires no expensive pre-computation of harmonic potential, nor does it require node locations. We assume here that the sweep is performed with respect to such a given arbitrary function, that may be natural or artificial and it suffices to terminate the sweep at the minima.

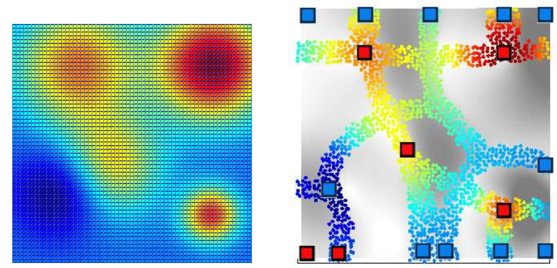
A disadvantage of the sweep method is that it pauses at the saddles. At a saddle where different components of the sweep merge into a single iso-contour, it is likely that one sweep component will arrive before the other, and active nodes will have to wait for the other component to arrive.

To overcome this issue, we sweep the network with the help of the Morse-Smale decomposition. That is, we sweep each cell of the network independent of the others. Since Morse-Smale cells do not contain saddle points, idle pauses for synchronization of sweep are not necessary. This saves much of the active time of nodes and speeds up the aggregation. The sweep proceed in several independent components in segments around the network. This parallelism greatly decreases the overall aggregation time, and reduces MAC layer collisions and interference. Quantitative ideas of these benefits will be described in the simulations section.

C. Topology faithful compression and signal reconstruction

We briefly discuss the signal compression value of summary information obtained from Morse-Smale decomposition. Note that the properly labeled dual complex itself encodes most of the topological properties of the signal. It represents all saddles, maxima and minima. The *contour tree* and related information can be extracted from this data, making it possible to extract high level routes, as described above. The complex in addition also encodes topological properties of the network itself, in particular presence of holes, and their relation to the critical points.

In some cases however, it may be desirable to obtain slightly more geometric information about the signal. While the dual declares the presence of Morse-Smale cells that are guaranteed to be simple and homogenous, it says nothing about the shape of these cells. The idea in this section is that in a network with locations, some geometric aspects of the decomposition can be obtained at a low cost. This can be done simply by tracing the cut locus, that is, the boundaries of cells and the maxima and minima. Given a path along the boundary, whose node locations are known, we now have the network decomposed into cells of known boundaries, that are guaranteed to be Morse-Smale cells. Therefore any interpolation that satisfies the properties of such a cell will represent a topology faithful reconstruction.



(i) (ii)

Fig. 4. Reconstruction. (i) original signal. (ii) Reconstruction information from paths along the cut locus, encompassing critical points of the signal. Maxima and minima are shown as red and blue points respectively.

Figure 4 shows the reconstruction information available from such a boundary extraction. For each cell, we simply represent each of the connected components of its boundary by a single path. A network of about 4000 nodes, in this case is represented by chains of cut locus of only 300 node locations and values. The interesting feature of the reconstruction information is that all the *important* points of the original signal have been represented.

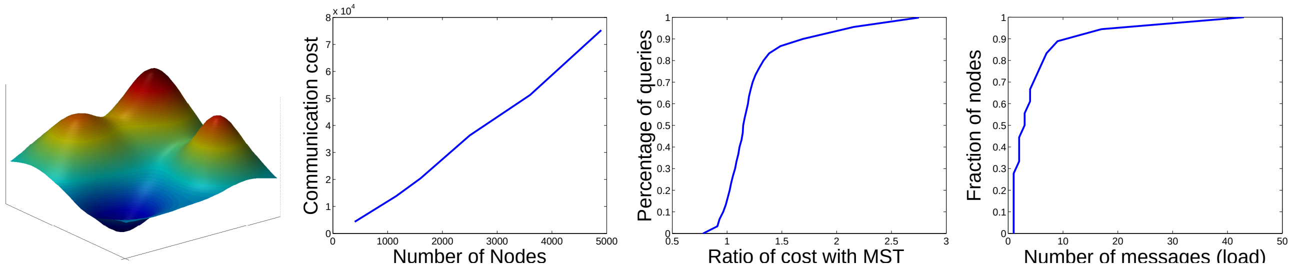
III. Simulations

We present here simulation results demonstrating performance of the routing and aggregation applications when executed on top of the Morse-Smale decomposition.

Computation of Morse-Smale Decomposition and Iso-contour Queries. We compare the performance of iso-contour queries with the contour tree approach [5]. The signal function and networks used in this section are identical to those in [5]. These networks do not have holes, since the contour tree construction does not handle holes well. Our algorithms however would also operate successfully on networks with non-trivial topologies.

We first sampled the function in Figure 5(i) with networks of varying number of nodes. The result in Figure 5(ii) shows that the communication cost for the Morse-Smale decomposition is linear in network size and therefore scalable.

Next, in a network of 1600 nodes, we ran iso-contour queries, from nodes chosen uniformly randomly from the network, and querying for a random signal value between the highest and lowest in the network. As in [5], we compare the query cost with that of a centrally computed minimum spanning tree, and plot the CDF of this ratio. Figure 5(iv) shows the CDF of the loads on network nodes. These graphs

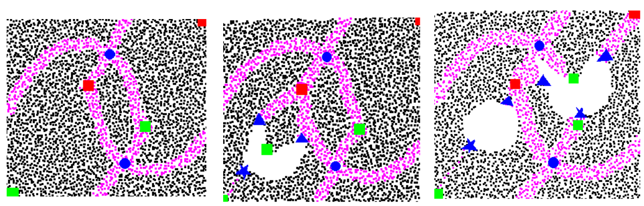


(i) (ii) (iii) (iv)

Fig. 5. (i) The continuous signal sampled by sensors; (ii) Message complexity of Morse-Smale decomposition; (iii) CDF of ratio of query cost to cost of MST (iv) CDF of node load distribution.

are found to be comparable or better than those of the contour tree method.

Aggregation by sweep. We described in the previous section why the decomposition can improve the efficiency of data aggregation by sweep. In this section, we describe simulation results that confirm our claim. Sweeps were simulated on the networks shown in fig. 6. An elementary medium access model



(i) (ii) (iii)

Fig. 6. Networks used in sweep. Around 4000 nodes, UDG, average node degree ~ 17. (i) without any holes; (ii) With One hole; (ii) With two holes.

with collisions and exponential back-off was used in these experiments. Time was measured by ticks of a global clock. We measured the total time taken to complete the sweep, average and max up-time per node, and the total number of medium access collisions. The up-time of a node was measured as starting at the tick when it received its first invitation to join the sweep, to the time it successfully sent out the final invitation to a lower neighbor. It measures how long a node needs to stay active during the sweep and correlates to its energy consumption. The results for the three networks above are presented in tables I, II and III respectively. We compare the performance of the sweep on the decomposition with that of the sweep without the decomposition, and with that of the sweep by a geographic coordinate (say X coordinate). The signal function is the one shown in figures **??**(i) & (ii).

Type	Total	Avg up-time	Max up-time	#collisions
Normal	12444	563.5	4376	9208
X coord	8602	311.3	992	8609
Decomposed	6927	288.0	1260	5056

TABLE I. The sweep time for the network without hole in fig 6(i).

Type	Total Time	Avg up-time	Max up-time	#collisions
Normal	12495	636.3	5893	8853
X coord	9060	335.3	2700	8381
Decomposed	6416	273.4	1591	5387

TABLE II. The sweep time for the network with 1 hole in fig 6(ii).

It is easy to see that the sweep on the decomposition outperforms all other schemes in almost all respects. This is to be expected, since the partitioning allows for more parallelism

Type	Total Time	Avg up-time	Max up-time	#collisions
Normal	17182	984.5	12147	10013
X coord	11748	579.1	8475	9156
Decomposed	7788	386.1	2056	5766

TABLE III. The sweep time for the network with 2 holes in fig 6(iii).

and less collisions. It avoids sweeps meeting at saddle points, and thus prevents a fast moving sweep frontier piece from waiting a long time for other contour pieces to catch up. The effect is even more pronounced in complex network topologies with holes. See table III where the maximum up-time of a node is many times smaller in the sweep on the decomposition.

IV. Conclusion

We developed a distributed algorithm to decompose a sensor field with respect to the sensor data so that each cell is simply connected with a homogeneous data flow. The philosophy of exploiting the sensor data characteristics in implementing networking operations is a promising direction in sensor network research.

References

[1] M. D. Carmo. *Riemannian Geometry*. Birkhäuser, Boston, Basel, Berlin, 1992.

[2] Q. Fang, J. Gao, L. Guibas, V. de Silva, and L. Zhang. GLIDER: Gradient landmark-based distributed routing for sensor networks. In *Proc. of the 24th Conference of the IEEE Communication Society (INFOCOM)*, volume 1, pages 339–350, March 2005.

[3] J. Gao, R. Sarkar, and X. Zhu. Morse-smale decomposition, cut locus and applications in wireless sensor networks. http://www.cs.sunysb.edu/˜rik/papers/morse-boundary-theory.pdf, 2008.

[4] J. W. Milnor. *Morse Theory*. Princeton University Press, Princeton, NJ, 1963.

[5] R. Sarkar, X. Zhu, J. Gao, L. J. Guibas, and J. S. B. Mitchell. Iso-contour queries and gradient descent with guaranteed delivery in sensor networks. In *Proc. of the 27th Annual IEEE Conference on Computer Communications (INFOCOM'08)*, pages 1175–1183, May 2008.

[6] P. Skraba, Q. Fang, A. Nguyen, and L. Guibas. Sweeps over wireless sensor networks. In *IPSN '06: Proceedings of the fifth international conference on Information processing in sensor networks*, pages 143–151, 2006.

[7] S. Smale. Morse inequalities for a dynamic system. *Bulletin of the AMS*, 66:43–49, 1960.

[8] R. Thom. Sur une partition en cellules associée à une fonction sur une variété. *Comptes Rendus de l'Académie de Sciences*, 228:973–975, 1949.

[9] Y. Wang, J. Gao, and J. S. B. Mitchell. Boundary recognition in sensor networks by topological methods. In *Proc. of the ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom)*, pages 122–133, September 2006.