

# Validating Sensors in the Field via Spectral Clustering Based on Their Measurement Data

H. T. Kung

Dario Vlah

{htk, dario}@eecs.harvard.edu

Harvard School of Engineering and Applied Sciences  
Cambridge, MA 02138

## ABSTRACT

*In this paper we introduce a spectral-based method for validating sensor nodes in the field via clustering of sensors based on their measurement data. We formalize the notion of peer consistency in measurement data by introducing a notion called “sensor indexing” and model the problem of identifying bad sensors as a problem of detecting peer inconsistency. Suppose all sensors have peers. Then by examining a certain number of leading eigenvectors of the measurement data matrix, we can identify those bad sensors which are inconsistent to peer sensors in their reported measurements. Further, we show that by deemphasizing or removing measurements obtained from these bad sensors we can improve the performance of sensor-based applications. We have implemented this spectral-based peer validation method and measured its performance by simulation. We report the effectiveness of the method in identifying bad sensors, and demonstrate its use in deriving accurate solutions in a localization application.*

## I. INTRODUCTION

In a distributed sensor application, the goodness of a sensor in the field may change due to a number of reasons. For example, sensors may malfunction or become damaged, get blown by the wind to a position where radio reception is poor, enter into a function-reducing mode due to low battery power, or even be maliciously compromised. This means that we may need to validate sensors periodically, in order to identify those which no longer function as expected. Then in the associated sensor-based applications, we can discount or remove measurements from these bad sensors.

In many circumstances it would be impractical to bring calibration instruments to the field for conducting sensor validation procedures on the spot. Fortunately, in a distributed sensor application, there are potentially many sensors in the environment, so they can check each other’s validity.

In this paper, we present a spectral clustering approach to validating sensors by peers in the field. We show that by clustering sensors based on their measurement data we can identify bad sensors with a high degree of reliability.

Once we remove these bad sensors, we end up with a smaller sensor set and also more appropriate sensor data for applications. Consequently, the sensing results may become more accurate while eliminating the wasted computing and communication with bad sensors. Furthermore, this could lead to a stealthier sensing environment and also improved protection against malicious tampering. Our approach of validating sensors in the field, followed by proper deemphasis or removal of measurement data from bad sensors departs from other approaches in distributed sensor applications where no attempts are made to remove bad sensors or discount their data, and emphasis is instead placed on being tolerant against erroneous measurements from bad sensors (e.g., those in localization with MDS [1] and SISR [2]).

At the heart of our approach is a new formulation, captured by the notion of “sensor indexing,” that allows us to express formally measurement consistency expected from “peer” sensors. We will first describe this formulation and give an overview on our peer-based approach of identifying bad sensors in Section II. In Section III we will describe the basic idea of spectral-based clustering and illustrate it with a simple example. In Section IV we will present a one-dimensional (1D) application of identifying bad sensors using 1D sensor indexing. Then, in Section V we will present a two-dimensional (2D) localization application using a 2D sensor indexing, along with some simulation results on performance improvements resulting from discounting identified bad sensors. We will conclude in Section VI.

## II. MODELING EXPECTED MEASUREMENT CONSISTENCY OF SENSORS

In this section, we formalize the expectation that those sensors which have similar sensing and system characteristics and are situated in close proximity with a similar environment will report similar measurement data. To this end, we use “sensor indexing” to capture these similarities in sensors. More precisely, we summarize below some concepts, assumptions and terms used in this paper.

- There are a number of *reference targets*, or simply *targets*, on which sensors report measurements. We

call the measurements of a sensor on a target the *weight* for this sensor-target pair.

- Sensors can be clustered based on their measurements so a cluster contains those sensors which report similar measurements. We call these clusters *measurement clusters*.
- Sensors can be mapped into an index space. *Peer sensors* or simply *peers*, are those which have nearby indices according to some distance metric for index pairs.
- Sensors can be clustered based on their indices so a cluster contains sensors with nearby indices, i.e., peer sensors. We call these clusters *peer clusters*.
- It is assumed that the indexing scheme satisfies the property that peer sensors will report similar measurements on the targets. Moreover, for any given measurement cluster, there is a corresponding peer cluster, and vice versa. (Sections III, IV and V describe some 1D and 2D indexing schemes.)
- It is assumed that each sensor has sufficiently many peers.
- A sensor whose measurement and peer clusters coincide is called a *good sensor*. Otherwise the sensor is called a *bad sensor*.

Choosing a proper granularity of measurement clusters (and the corresponding peer clusters) is a design choice. One may divide a measurement cluster into smaller ones, by recognizing finer measurement differences among sensors. (The method of Section IV-A examines additional leading eigenvectors for finer measurement clustering). Generally speaking, based on finer-grain measurement clusters, we can increase the success rate of catching bad nodes, but possibly at the expense of more false positives. After a certain point, decreasing the cluster size will have diminishing returns. When this point is reached, we may use other methods of identifying additional bad sensors without decreasing cluster sizes. For example, in Sections IV and V we identify sensors on the edge of a measurement cluster as bad sensors.

To illustrate this concept of sensor indexing, we consider a simple example described in [3], where the sensor property utilized as the sensor index is the sensors' antenna polarization angle. In this case, peer sensors are those which have similar antenna polarizations. Suppose that the received signal strength (RSS) values reported by a sensor on some target correlate with the matching degree of the antenna polarizations of the sensor and the target. Then, those peer sensors which are in good working conditions are expected to report similar RSS measurements on the same targets.

Like granularity of measurement clusters, choosing a proper sensor indexing scheme is a design choice. For a given sensor-based application the indexing scheme

should be related to sensing metrics used so peer sensors can be expected to perform similarly for the application. For example, for a 2D localization application using RF ranging measurements, it would be appropriate to index sensors based on their 2D geographical locations.

Next, we describe our spectral-based method for forming measurement clusters of sensors and how we use these sensor clusters to identify bad sensors. We will first form a *measurement matrix*  $\mathbf{A}$  where entry  $a_{ij}$  of  $\mathbf{A}$  is the measurement weight for the sensor-target pair  $(s_j, t_i)$  defined above. We will then perform spectral analysis on  $\mathbf{A}^T\mathbf{A}$ , since the eigenvectors of  $\mathbf{A}^T\mathbf{A}$  characterize similarities among sensors in their measurement data. Based on the eigenvectors of  $\mathbf{A}^T\mathbf{A}$ , we will cluster sensor nodes using methods such as those described in Section IV-A.

From the above, it follows that, to identify bad sensors, we will form the measurement cluster to which a sensor belongs, and then determine if the measurement cluster agrees with the corresponding peer cluster. We assume that there are plentiful sensors so that each sensor has a sufficient number of peers for the bad sensors identification method. It follows from the above discussion that a sensor is deemed to be bad if the sensor satisfies one of the two conditions:

- (C1) the sensor is in a small unique measurement cluster, or
- (C2) the sensor is in a small out-of-place component (in the index space) of a large measurement cluster.

Therefore the essence of the clustering problem studied in this paper is to identify these two conditions for sensors. In the rest of the paper, we show that we can achieve this objective by examining leading eigenvectors of  $\mathbf{A}^T\mathbf{A}$ .

### III. EXAMPLE OF SPECTRAL CLUSTERING OF SENSORS

To illustrate our approach of computing measurement clusters of sensors based on spectral clustering we present a simple example where sensors and targets are placed on a line, and sensors report RF measurements on the targets. We deliberately simplify the problem to permit easy visualization and interpretation of clustering results, while still capturing essential issues of interest. When extending this result to real-world applications, various details may need to be added, such as more detailed path loss models or antenna radiation patterns.

In this example there are 19 sensors and 3 reference targets, numbered 0-18 and 0-2, respectively. Further, sensor  $i$  and target  $j$  are at locations  $5i$  and  $45j$ , respectively. We model sensor measurements of targets as a function of the location difference; that is, the measurement of target  $j$  obtained at sensor  $i$  is a function of the difference between their location values. In particular, we use a step function shown in Figure 1. (Use of a step function makes



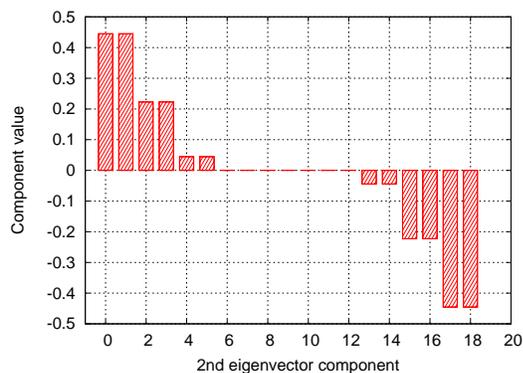


Fig. 4: The second eigenvector of  $\mathbf{A}^T \mathbf{A}$  with values of its elements (also called components) shown. Note that components here correspond to sensor nodes.

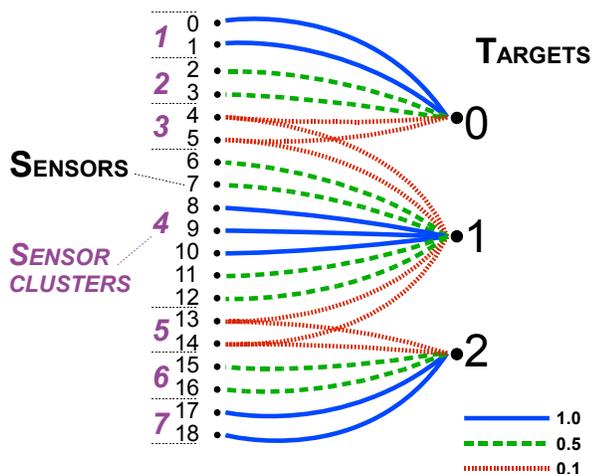


Fig. 5: Bipartite sensor-target measurement graph, where color-coded scores are blue = 1, green = 0.5 and red = 0.1. The seven sensor clusters shown are those found by the second leading eigenvector of Figure 4. Note that with the help of the principal eigenvector of Figure 3, cluster 4 will be correctly partitioned into the two clusters  $\{8, 9, 10\}$  and  $\{6, 7, 11, 12\}$ .

$\{11, 12\}$  should form two clusters. One can check that this partition of  $\{6, 7, 8, 9, 10, 11, 12\}$  into these two clusters is actually implied by the principal eigenvector of Figure 3, as the two groups attain two distinct values 0.25 and .5. Thus, for this example, by using both the principal and second eigenvectors together, we have correctly obtained all the eight clusters:  $\{0, 1\}$ ,  $\{2, 3\}$ ,  $\{4, 5\}$ ,  $\{8, 9, 10\}$ ,  $\{6, 7, 11, 12\}$ ,  $\{13, 14\}$ ,  $\{15, 16\}$ , and  $\{17, 18\}$ .

We have seen in this example that the first and second eigenvectors complement each other, in the sense that each can fix the alias problems of the other. This phenomenon is

not an accident. Without loss of generality, we consider the case that eigenvectors are computed by the power method. Note that because the matrix  $\mathbf{A}$  has non-negative entries, using a positive initial vector for the power method results in the principal eigenvector having aliased values when the underlying graph is a symmetric bipartite graph, as in Figure 5. Being orthogonal to the principal eigenvector, the second one must contain both positive and negative components. As a result, the second eigenvector, computed by the power method starting with an initial vector of both positive and negative values, does not suffer from the same alias problems as the first one. However, due to cancellation resulting from combining positive and negative values, the second leading eigenvector may suffer from other alias problems, which the first one does not share since all its values are positive. This means that the principal and second eigenvectors can fix each other's aliases. (We note that the use of leading eigenvectors in this manner is a technique also known to multidimensional scaling, a statistical method for computing data similarity. [6])

Using this joint approach of principal and non-principal eigenvectors, we have a general method of automatically finding measurement clusters of sensors from the measurement matrix.

#### IV. APPLICATION 1: IDENTIFYING BAD SENSORS USING 1D INDEXING

To illustrate our approach of using spectral clustering for identifying bad sensors, we consider a model 1D scenario. There are 80 sensors on a line performing RF measurements of each other; thus, the set of sensors themselves also serves as the set of reference targets. A sample placement with four 20-node clusters is depicted in Figure 6. Among this set of sensors we wish to identify the bad ones. We will index the sensors by their 1D locations computed by some localization algorithm, making this an example of 1D indexing.

We use a standard path loss function,  $1/d^4$ , to model the measurement of a target by a sensor, where  $d$  gives the distance from the sensor to the target. Furthermore, to model the measurement errors we modify the path loss by a random factor  $\alpha \sim \mathcal{N}(\mu, \sigma)$ . We denote the distribution parameters  $\mu$  and  $\sigma$  as either  $\mu_{good}, \sigma_{good}$ , or  $\mu_{bad}, \sigma_{bad}$ , representing good and bad links, respectively. Given the sensors' ground truth positions we generate a measurement for each pair of sensors, and form the measurement matrix  $\mathbf{A}$ . For instance, Figure 7 shows the subset of measurements obtained by just one sensor.

As described in Section III we will examine a certain number of leading eigenvectors of  $\mathbf{A}^T \mathbf{A}$  to identify measurement clusters of sensors in terms of their measurement similarities with respect to the targets. The number of leading eigenvectors needed is related to the number of leading



Fig. 6: Depiction of a 1D sensor placement scenario with 4 clusters of 20 sensors each.

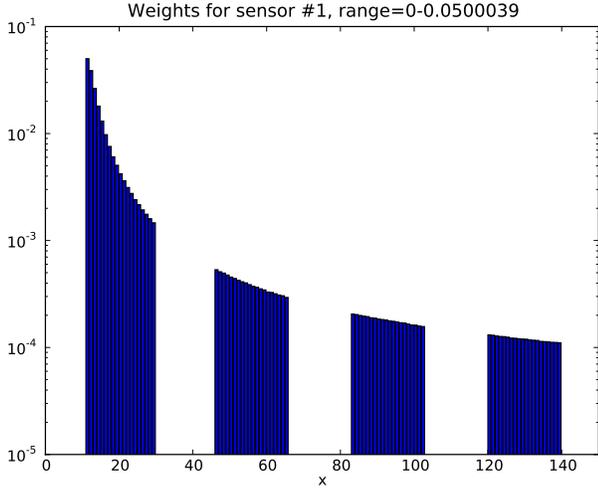


Fig. 7: The signal strength measurements taken at sensor 1 of all other sensors placed in four clusters. These measurements constitute the first row of the measurement matrix  $\mathbf{A}$ .

eigenvalues of  $\mathbf{A}^T \mathbf{A}$  which are significantly larger than the rest. For example, for the weight matrix  $\mathbf{A}$  corresponding to Figure 6, assuming no error, the leading four eigenvalues of  $\mathbf{A}^T \mathbf{A}$  are 117, 104, 97 and 93, while the rest are 38.6 or smaller. Matrix  $\mathbf{A}^T \mathbf{A}$  has four significant leading eigenvalues due to the presence of the four clusters in the sensor placement. Three of the corresponding leading eigenvectors of  $\mathbf{A}^T \mathbf{A}$  are depicted in Figure 8.

#### A. Sign-based Spectral Clustering

Building on our arguments in Section III, we now describe a sign-based spectral clustering, which will provide a straightforward way to compute measurement clusters of sensors given a set of leading eigenvectors of  $\mathbf{A}^T \mathbf{A}$ . Suppose that we use the  $k$  leading eigenvectors. Then multiple sensors will be clustered together if their corresponding components in these  $k$  eigenvectors have matching signs. (Note that  $k$  leading eigenvectors can specify no more than  $2^{k-1}$  clusters, given that the principal eigenvector always contains components of the same sign.) As an example, consider the sign-based process of clustering sensors in Figure 6 with  $k = 3$ ; the 3 leading eigenvectors are in turn shown in Figure 8. Indeed, we can see that the component signs over the 3 eigenvectors are unique for each of the four sensor clusters identified. It is expected that the sign-based clustering generally gives coarse clustering, since only signs of component values rather than their actual values are used.

In practice, sign-based clustering is robust against those sensor measurements which may have significant variation. In this case, component values of eigenvectors will have large variance so their precise values cannot be used in determining clustering. Sign-based clustering is a method to address this uncertainty in sensor measurements.

#### B. An Illustrative Sensor Validation Example

In this section, we first illustrate how our spectral clustering method can identify a bad sensor for the model 1D application. We then discuss some considerations in designing a spectral clustering scheme for detecting bad sensors.

Suppose that one of the sensors in Figure 6 malfunctions, and obtains measurements of other sensors polluted with noise  $\alpha \sim \mathcal{N}(\mu_{bad}, \sigma_{bad})$  larger than for the other sensors. An instance of the weights from just sensor 1 to all other sensors, including the bad sensor, are shown in Figure 9.

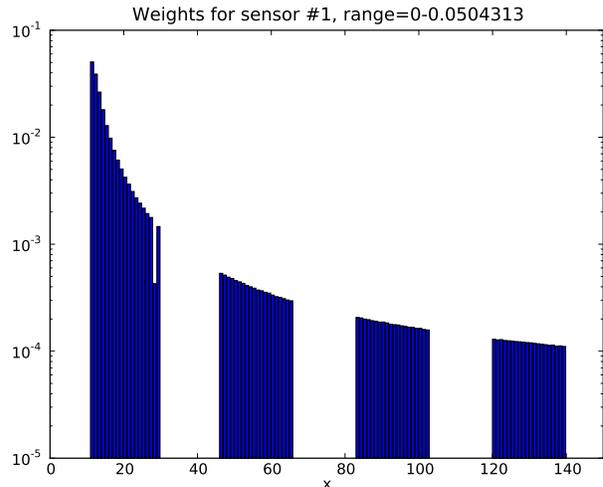


Fig. 9: The signal strength measurements taken at sensor 1 of all other sensors placed in four clusters, where one of the other sensors (around  $x=27$ ) is malfunctioning.

Let  $\mathbf{A}$  be the measurement matrix including bad measurements of the bad sensor. Applying the sign-based spectral clustering method of Section IV-A to the  $\mathbf{A}^T \mathbf{A}$ , we obtain measurement clusters shown in Figure 10b.

We assume that the sensors do not know their own locations; instead, they must compute them via localization from the ranging measurements. Figure 10a shows the ground truth locations and the position of the malfunctioning sensor, while Figure 10b shows the result of localization using the MDSR [6] algorithm.

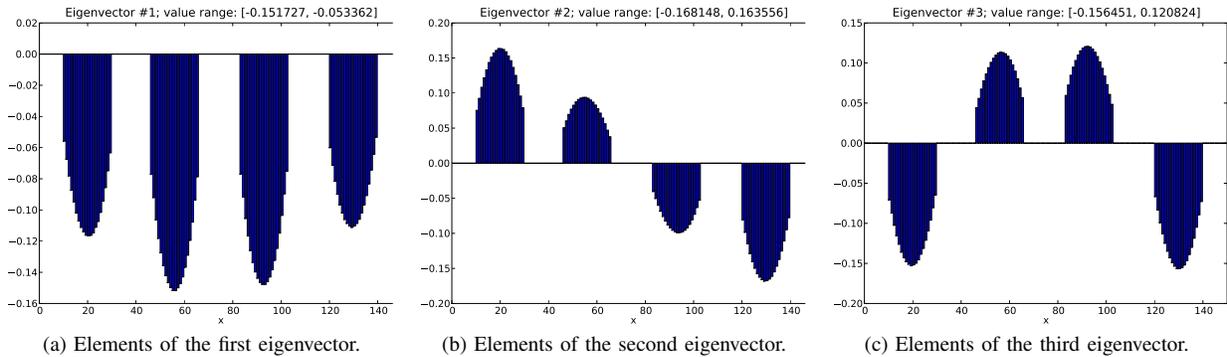


Fig. 8: Elements of the 3 leading eigenvectors of  $\mathbf{A}^T \mathbf{A}$  for the sensor placement of Figure 6.

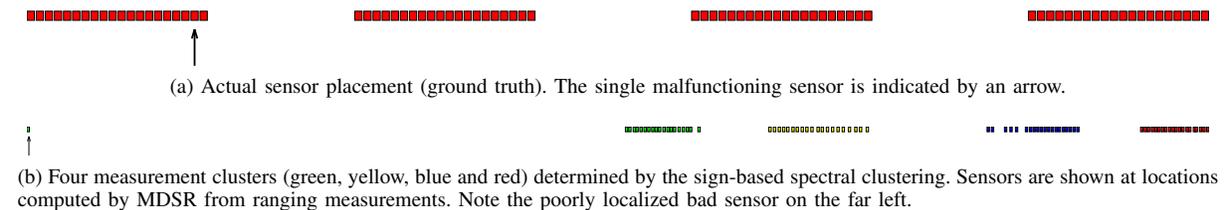


Fig. 10: Depiction of a 1D sensor placement scenario with 4 clusters.

Let us use the computed sensor location as the index. Given the computed sensor locations and measurement clusters shown in Figure 10b, we now apply conditions C1 and C2 of Section II to classify bad sensors as follows. For condition C1, we deem a sensor bad if it resides in a standalone cluster smaller than a certain threshold size  $S_{min}$ . For condition C2, we classify a sensor as bad if it is one of the  $\tau$  % sensors furthest away from the centroid of the measurement cluster which contains the sensor, where  $\tau$  is some small threshold. As we can see in Figure 10b, the bad sensor indicated by the arrow is at the edge of its cluster, so it can be correctly identified even with a minimal  $\tau$ . Note that the distance metric used here is that of the index space as described in Section III. In this example, the distance for a pair of sensors is the absolute difference of their locations computed by MDSR.

This specific way of detecting bad sensors based on condition C2 relies on the following two assumptions. First, measurement clusters are stable under a few bad sensors. Second, caused by measurement errors the computed location of a bad sensor is normally pushed to the edge of its measurement cluster in the index space. We will have further discussion on this in Section V.

### C. Simulation Results on Large Systems

In this section, we present simulation results on the model application with 400 sensors. We assume that the sensors are evenly partitioned into four groups of 100. We further assume that some randomly selected sensors are bad sensors, in the sense that all their measurements

of other sensors have a random measurement error  $\alpha \sim \mathcal{N}(\mu_{bad}, \sigma_{bad})$ . That is, if the true measurement for a sensor-target pair is  $w$ , then the observed one is  $\alpha w$ . We used the error parameters  $\mu_{bad} = 10$  and  $\sigma_{bad} = 3.2$ . For all other links where both transmitting and receiving sensors are good, we omitted the error term; that is,  $\mu_{good} = 0$  and  $\sigma_{good} = 0$ . The purpose of the simulation is to assess the effectiveness of our spectral clustering method in identifying these bad sensors. In the simulation, there are 50 bad sensors.

Let  $B$  be the ground truth number of bad sensors input to the simulator,  $T$  the total number of sensors deemed as bad sensors by our clustering method, and  $B^*$  the number of sensors that the clustering method correctly identified as bad. Hence,  $B^* \leq B$  and  $B^* \leq T$ . The higher the detection rate  $B^*/B$  is, the better the method is. Note that  $T - B^*$  is the number of false positives. We are interested in raising the *detection rate*  $B^*/B$ , without raising the false positive rate  $(T - B^*)/T$  significantly. But usually there is a tradeoff between the two factors.

We study performance under varying numbers  $k$  of leading eigenvectors used in the spectral clustering method. In the simulator we assume that the minimum cluster size  $S_{min}$  from Section IV-B is 2. Meanwhile, we explore a range of values for the parameter  $\tau$ , or the fraction of sensors furthest from a cluster center which we will declare bad. Increasing  $\tau$  leads to identifying more bad nodes, but usually at the expense of more false positives.

The following observation supported by simulation results in the rest of this section (see Figure 13) and in

Section V, is worth mentioning. Before a certain threshold (e.g., 6) on the number  $k$  of leading eigenvectors used in spectral clustering is reached, when we increase  $k$ , we can catch more bad sensors while keeping the  $\tau$  value constant. This means that by increasing  $k$ , we can increase the bad sensor detection rate, without increasing the false positive rate. This result follows from the fact, noted earlier in Section III, that use of more eigenvectors (before reaching some threshold on  $k$ ) will give finer-grained measurement clusters, which in turn will allow us to identify bad sensors which have smaller errors in their computed locations.

Figures 11 and 12 show results on two performance metrics: the bad sensor detection rate and false positive rate, respectively, with respect to the  $\tau$  threshold. Additionally, Figure 13 shows the effect of the number  $k$  of leading eigenvectors used on detection and false positive rates. We note the following from the figures:

- 1) When  $k$  increases, the detection performance improves for each value of  $\tau$ . When  $k$  is at its highest value of 14, the best performance is achieved (see Figure 11).
- 2) For certain values of  $k$ , the false positive rates as a function of  $\tau$  exhibit a minimum; in Figure 12 these minimums are reached around  $\tau = 10\%$  to  $15\%$ .
- 3) For a fixed  $\tau$ , there are values of  $k$  which give minimal false positive rates; for example, for  $\tau = 10\%$ , the lowest false positive rate in Figure 13 is reached for  $k = 6$ .
- 4) Increasing  $k$  past that giving the minimum false positive rate only provides a minor increase in detection rate. Given that the false positive rate increases rapidly, the best value for  $k$  is close to the number of major sensor clusters, or, equivalently, the number of leading eigenvalues significantly larger than the rest.

## V. APPLICATION 2: 2D LOCALIZATION WITH 2D SENSOR INDEXING

After having described in the previous section the method of detecting bad sensors using 1D indexing, we now have enough context and terminology to describe Application 2, where we extend the method to 2D indexing and show its utility in improving the performance of a 2D localization application.

We again assume that there are  $N$  ranging sensors which take measurements of received RF signal strength at each other, but are now placed in a 2D space. As in Application 1, we will form the measurement matrix and use spectral clustering methods to identify bad sensors whose measurements are inconsistent with those of peer sensors.

Since received signal strength reflects geographical locations of sensors, it would be natural to define the 2D index of a sensor to be its geographical 2D location. But

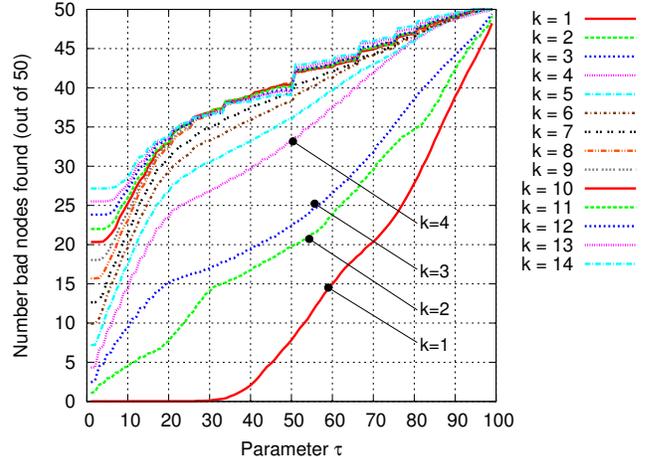


Fig. 11: Number of bad sensors, out of 50, identified for increasing values of parameter  $\tau$  and a range of values for the number of leading eigenvectors  $k$  used for spectral clustering.

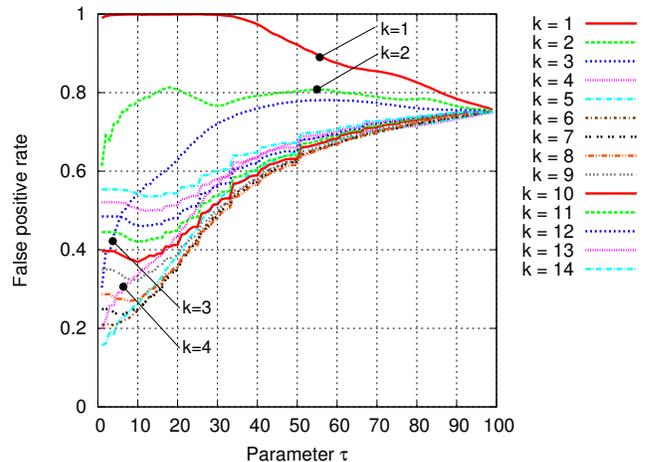


Fig. 12: False positive rate of the spectral clustering method under increasing threshold  $\tau$ , for a range of number  $k$  of leading eigenvectors used for spectral clustering.

we cannot assume that we know sensors' 2D locations, for otherwise we would not need to do the localization application in the first place. Also, if we know the 2D locations of the sensors, we could have simply used this location information to check if signal strength measurements are compatible with transmission distances and, accordingly, validated or invalidated the associated sensors.

Instead, like the 1D indexing in Section IV, we will define the 2D index of a sensor to be its *computed* 2D location, obtained from a localization method (e.g., MDSR) using the given measurement matrix as input. Consider a bad sensor,  $X$ , which belongs to a measurement

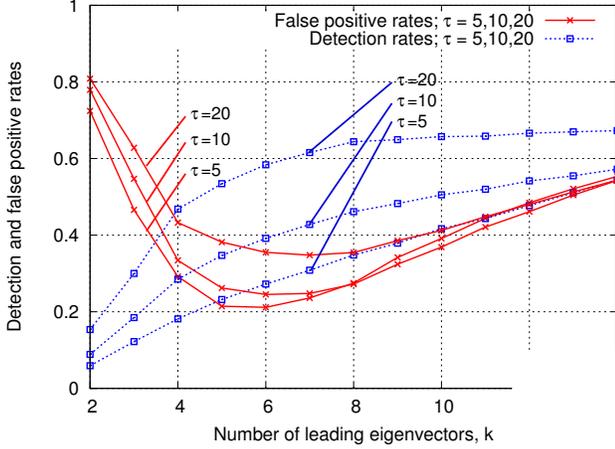


Fig. 13: Detection rate and false positive rate against number  $k$  of leading eigenvectors used clustering for three representative values of  $\tau$ .

cluster of sensors. Suppose that most of the nodes in the measurement cluster are good. Being in a measurement cluster, these good sensors will have measurements with respect to sensors outside the cluster. Thus we can expect that a localization method that places these good sensors close to each other, resulting in a tight cluster in the index space. But for a bad sensor  $X$ , being a bad sensor, the computed location tends to be pushed away from the good nodes. Thus in the index space, the location of  $X$  is likely to be on the edge of the measurement cluster containing  $X$ .

By this reasoning, our method of identifying bad sensors for the 2D localization application works as follows:

- 1) Based on the measurement matrix, localize good sensors using some standard localization method.
- 2) Each sensor is given a 2D index which is its computed location obtained from Step 1.
- 3) Based on the measurement matrix, compute measurement clusters of sensors using a spectral clustering method as described in Section IV-A.
- 4) Identify bad sensors by examining how measurement clusters are distributed in the 2D index space of sensors. More specifically, check conditions C1 and C2 of Section II; for this purpose, we again employ the techniques from Section IV-B using thresholds  $S_{min}$  and  $\tau$  to catch bad sensors.

It is important to note that for the purpose of providing sensor indexing, the localization method for step 1 is only required to preserve measurement clusters for good nodes. Other metrics, such as absolute accuracy of computed locations are not important. Thus computed measurement clusters could be displaced, rotated or flexed from their ground-truth locations. Well-known localization methods,

such as MDS [1], normally perform well under this relaxed criterion, and thus can be successful in providing the 2D sensor indexing.

#### A. Simulation Results on Using Measurement Clusters of Sensors in Identifying Bad Sensors

We performed a simulation study on the effectiveness of using measurement clusters of sensors in identifying bad nodes. We created four groups of 100 nodes in the corners of a square region. Within each group the 100 sensors had locations chosen uniformly at random, resulting in node

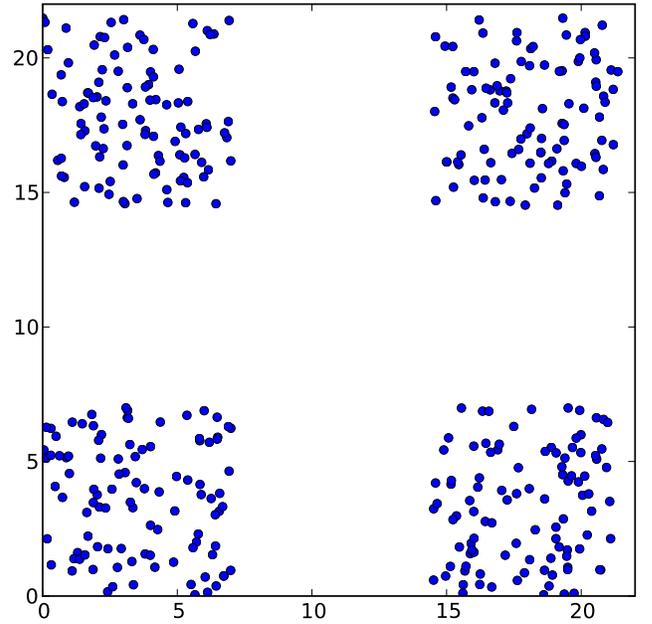


Fig. 14: 100 nodes each are placed uniformly at random into  $7 \times 7$  regions in each corner.

From these ground truth locations we computed synthetic RF ranging measurements using an empirical two-ray propagation model trained on real-world data [2]. Then, we chose a random sample of  $B = 50$  sensors to be bad nodes. The error behavior of bad nodes was identical to that in Section IV, that is, measurements in both directions involving a bad node would be attenuated by a random variable  $\alpha \sim \mathcal{N}(10, 3.2)$ . Such errors represent some persistent effect such as the attenuation due to lossy cables or antenna connectors, shadowing, etc. The ground truth measurements combined with bad node errors constituted the measurement matrix  $\mathbf{A}$ .

We performed the 4 steps of identifying bad nodes listed earlier in this section as follows.

- 1) Based on the measurement matrix  $\mathbf{A}$ , we computed relative sensor locations using the MDS [1]

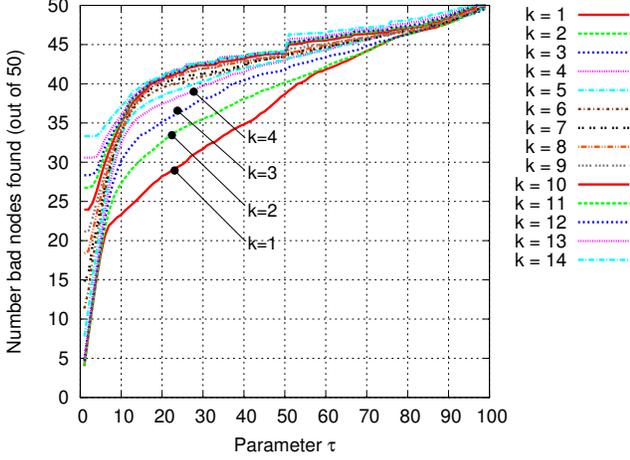


Fig. 15: Number of bad sensors, out of 50, identified for increasing values of parameter  $\tau$  and a range of values for the number  $k$  of leading eigenvectors used for spectral clustering.

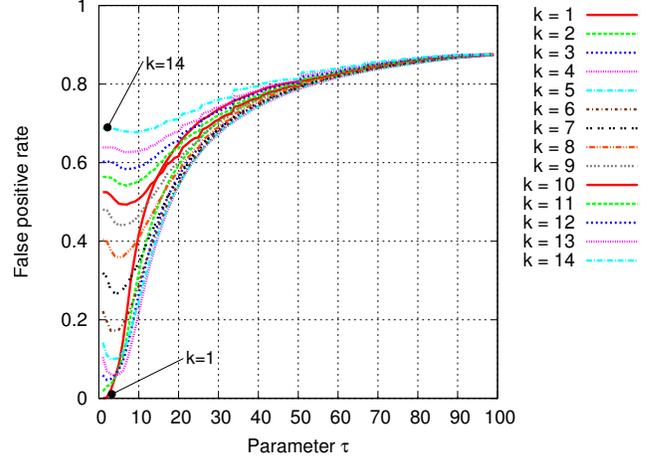


Fig. 16: False positive rate of the spectral clustering method under increasing threshold  $\tau$  for a range of number  $k$  of leading eigenvectors used for spectral clustering.

method. MDSR operates in two steps; first, a classic multi-dimensional scaling (MDS) step [6] computes an approximate solution from the measurement matrix  $\mathbf{A}$ . This step requires no initial solution. The outcome of MDS then serves as an initial solution for the second step, least squares optimization (LSQ)<sup>1</sup>, which produces a more accurate final result. For the purpose of indexing, we only performed the first, MDS step.

- 2) The computed location of each sensor was used as its 2D index.
- 3) Using sign-based spectral clustering on matrix  $\mathbf{A}^T \mathbf{A}$ , we computed measurement clusters of sensors. We varied the number of eigenvectors  $k$  used in this process between 1 and 14.
- 4) We identified bad sensors as those sensors which are either (1) in a measurement cluster of size 1, or (2) within each measurement cluster, in the subset of  $\tau\%$  of sensors furthest from the cluster centroid. We varied the parameter  $\tau\%$  between 0 and 100%.

We measured the performance of each 4-step run using similar metrics as in Section IV-C, that is, the detection and false positive rates. We present performance results in these metrics in Figures 15 and 16 with respect to the  $\tau$  parameter. Additionally, in Figure 17 we plot results on the same metrics with respect to  $k$ .

The most notable property of these results are the large increases in the number of bad sensors detected as  $\tau$

<sup>1</sup>In fact, MDSR performs a weighted least-squares optimization, but for the results reported in Table I, we chose not to use it since it performed worse than unweighted LSQ on our input.

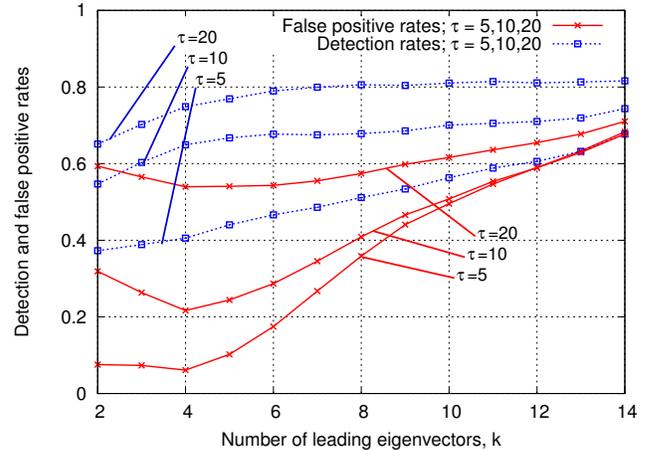


Fig. 17: Detection rate and false positive rate plotted against number  $k$  of leading eigenvectors used for spectral clustering for three representative values of  $\tau$ .

increases in the region  $\tau \leq 20$ , especially for the first few smallest values of  $k$  (see Figure 15). This indicates that the density of bad nodes is indeed larger on the edge (in the index space) of the measurement clusters. Furthermore, just like in the 1D scenario of Section IV, Figure 17 shows that, at first, increasing  $k$  decreases the false positive rate; the reason is that the clustering with too few eigenvectors generates too few clusters, and consequently fails to expose some cluster border areas with high densities of bad nodes. Finally, increasing the  $k$  parameter starts to increase the false positive rate once the bad node-rich border areas have been exhausted. The minimum false-positive rate is

again achieved near the number of significant eigenvectors induced by the clustering structure of the sensor placement.

The bad node classification can be useful to applications; for example, a localization algorithm might weight down those measurements associated with bad nodes to avoid excessive distortion of the results. We tested this process by modifying the LSQ step of MDSR to use weighted least squares optimization, where any distance constraints involving a bad node are weighted down by some factor  $\beta < 1$ ; we used  $\beta = 1/100$  in our simulations. We evaluated the performance of unmodified MDSR and the weighted MDSR (denoted WMDSR for short) at the known good nodes only; we report the outcomes in Table I.

#Bad Nodes	5	10	15	20	25
% Improvement WMDSR over MDSR	15.3	10.6	15.6	6.8	5.6

TABLE I: The percent decrease in localization error of MDSR of known good nodes achieved by WMDSR, where the nodes identified as bad by sensor clustering are weighted down. The values are averages over 10 runs. Results are reported for the number of bad nodes increasing from 5 to 25.

## VI. CONCLUSION AND FUTURE WORK

We have presented a spectral clustering approach to validating sensors via their peers. The work is motivated by the fact that it is often difficult to use instruments to validate equipment in the field and, as a result, peer-based validations can be important in these situations. We have modeled the problem as a clustering problem in the sense that sensors in the same environment will be clustered together and will behave similarly as far as the reference targets are concerned, while a bad sensor must be on the edge (in the index space) of the same cluster, or in a different measurement cluster. Thus, we can identify bad sensors through the clustering structure.

A key result of this paper is on the use of leading eigenvectors in forming measurement clusters of sensors. Although spectral-based clustering is well known in the literature [7], our use in validating sensors in the field appears to be novel. More specifically, to the best of our knowledge, our formulation of sensor indexing and our method of identifying bad sensors using conditions C1 and C2 of Section II, are new. Our application of sign-based spectral clustering to finding measurement clusters of sensors also appears to be new. As shown in our simulation experiments involving 1D and 2D indexing, the method can identify bad sensors with high accuracy and low false positive rate. Furthermore, in the 2D indexing example, we have shown that the performance of the corresponding

sensor-based application improved when nodes invalidated by our method are weighted down.

The sensor validation methodology proposed in this paper is quite general, and opens up a number of avenues of future work. For instance, there is the question of which specific sensor properties to use for indexing. In addition to the straightforward ones such as those we used, there might be additional ones based on other sensor modalities, or certain transforms of the sensing data such as the Hough transform [8]. Secondly, there could be other potential clustering methods in addition to the spectral clustering we used in this paper. For this purpose we plan to evaluate methods such as k-means clustering, or spectral clustering using k-means to cluster eigenvectors. Finally, there are potentially many other sensor-based applications where the proposed sensor validation approach could be effective; we are exploring candidate applications such as cognitive radio spectrum allocation.

## ACKNOWLEDGEMENTS

This material is based on research sponsored by Air Force Research Laboratory under agreement numbers FA8750-08-1-0220, FA8750-09-2-0180 and FA8750-08-1-0191. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright notation thereon. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of Air Force Research Laboratory or the U.S. Government.

## REFERENCES

- [1] Y. Shang, W. Ruml, Y. Zhang, and M. Fromherz, "Localization from connectivity in sensor networks," *Parallel and Distributed Systems, IEEE Transactions on*, vol. 15, no. 11, pp. 961–974, 2004.
- [2] H. T. Kung, C. K. Lin, T. H. Lin, and D. Vlah, "Localization with Snap-Inducing Shaped Residuals (SISR): Coping with Errors in Measurement," in *The 15th Annual International Conference on Mobile Computing and Networking*, Beijing, China, Sep. 2009.
- [3] H. T. Kung and D. Vlah, "A Spectral Clustering Approach to Validating Sensors via Their Peers in Distributed Sensor Networks," in *Second International Workshop on Sensor Networks (SN2009)*, San Francisco, CA, Aug. 2009.
- [4] H. T. Kung and B. W. Suter, "A hub matrix theory and applications to wireless communications," *EURASIP Journal on Advances in Signal Processing*, vol. 2007, no. 1, pp. 60–60, 2007.
- [5] A. Berman and R. J. Plemmons, *Nonnegative Matrices in the Mathematical Sciences*. New York, NY: Academic Press, 1979.
- [6] T. F. Cox and M. A. A. Cox, *Multidimensional Scaling, 2nd Ed.* New York, NY: Chapman & Hall/CRC, 2001.
- [7] A. Y. Ng, M. I. Jordan, and Y. Weiss, "On spectral clustering: Analysis and an algorithm," in *Advances in Neural Information Processing Systems 14*. MIT Press, 2001, pp. 849–856.
- [8] R. O. Duda and P. E. Hart, "Use of the Hough Transformation to Detect Lines and Curves in Pictures," *Communications of the ACM*, pp. 11–15, Jan. 1972.