Nonparametric Change Detection and Estimation in Large Scale Sensor Networks

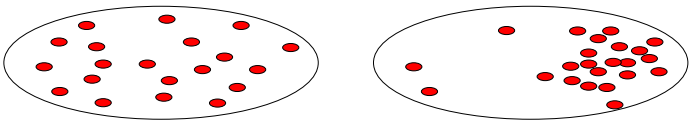
Ting He, Shai Ben-David, and Lang Tong†

***Abstract*— The problem of detecting changes in the distribution of alarmed sensors is considered. Under a nonparametric change detection framework, we present several detection and estimation algorithms based on the Vapnik-Chervonenkis theory. Theoretical performance guarantees are obtained by providing error exponents for false-alarm and miss detection probabilities. Recursive algorithms for the efficient computation of test statistics are derived. The estimation problem is also considered in which, after detection is made, the location with maximum distribution change is estimated.**

***Index Terms*— Nonparametric change detection, Sensor Networks, Detection and estimation algorithms.**

I. Introduction

We consider the detection of certain phenomenal change in a large-scale randomly deployed sensor field. For example, sensors may be designed to detect certain chemical components. When the sensor measurement exceeds certain threshold, the sensor is "alarmed". The state of a sensor depends on where it resides; sensors in some area are more likely to be in the alarmed state than others are. We are not interested in the event that certain sensors are alarmed. We are interested instead in whether there is a change in the geographical distribution of alarmed sensors from data collections at two different times. Such a change in distribution could be an indication of abnormality.



First data collection Second data collection

Fig. 1. Reported alarmed sensors (red) in two collections.

We assume that some (not necessarily all) of the alarmed sensors are reported to a fusion center, either through the use a mobile access point (SENMA [1]) or using certain in-network routing scheme. Suppose that the fusion center obtains reports of the locations of alarmed sensors, as illustrated in Fig. 1, from two separate data collections. In the ith report, let the location of alarmed sensors have some unknown distribution P_i, and each sample S_i be a set of locations drawn independently according to P_i. The change detection problem considered in this paper is one of testing whether $P_1 = P_2$ without making prior assumptions about the data generating distributions P_i. Note that P_i only specifies the geographical distribution of alarmed sensors. The joint distribution of alarmed and non-alarmed sensors is not specified completely. A change in P_i may be caused by the change of the actual phenomenon or the change of the sensor lay-out.

Such a general nonparametric assumption comes with a cost of usually requiring large sample size, which renders the solution in this paper most applicable in large-scale sensor networks where it is possible to obtain a large amount of sensor data.

There is also a related estimation problem in which, assuming that the detection of change has been made, we would like to know where in the sensor field the change has occurred, or where the change is the most significant (in a sense that will be made precise later).

A. Summary of Results

In this paper we present a number of nonparametric change detection and estimation algorithms based on an application of Vapnik-Chervonenkis Theory [2]. The basis of this approach has been outlined in [3] where we provided a mathematical characterization of changes in distribution. Our focus in this paper is on the algorithmic side, aiming at obtaining practical algorithms that scale with the sample size along with a certain level of performance guarantee.

We first present results that establish a theoretical guarantee of performance. The nonparametric detection problem considered here depends on the choice of the distance measure between two probability distributions, and the choice is usually subjective. We consider two distance measures in this paper. The first is the so-called $\mathcal{A}$*-distance* (also used in [3]) that measures the maximum change in probability on $\mathcal{A}$—a collection of measurable sets. The second is called relative $\mathcal{A}$-distance—a variation from that in [3]—for cases when the change in probability is concentrated in areas of small probability. With these two distance measures, we apply the Vapnik-Chervonenkis Theory to obtain exponential bounds[1]

†Corresponding author

Ting He and Lang Tong are with the School of Electrical and Computer Engineering, Cornell University, Ithaca, NY 14853. Email:{th255@, ltong@ece.}cornell.edu. Shai Ben-David is with the School of Computer Science, University of Waterloo, Waterloo, ON N2L 3G1, Canada. Email: shai@cs.uwaterloo.ca. This work was supported in part by the Multidisciplinary University Research Initiative (MURI) under the Office of Naval Research Contract N00014-00-1-0564, and Army Research Laboratory CTA on Communication and Networks under Grant DAAD19-01-2-0011, and the National Science Foundation under Contract CCR-0311055. Part of this work was presented in Conference on Information Sciences and Systems (CISS) 2004, Army Science Conference (ASC) 2004, and 2005 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP).

[1]Here we mean the error probabilities decay exponentially with the increase of sample size.

on detection error probabilities and establish the consistency results for the proposed detector and estimator.

Next we derive a number of practical algorithms. The complexity of applying the Vapnik-Chervonenkis Theory comes from the search among a (possibly infinite) collection of measurable sets. In particular, given data S being the union of the samples from the two collections, *i.e.*, $S = S_1 \bigcup S_2$, the key is to reduce the search in an infinite collection of sets (*e.g.*, planer disks) to a search in a *finite* collection $\mathcal{H}(S)$ (a function of S). Here we need a constraint on $\mathcal{H}(S)$ such that this reduction does not affect the performance.

We consider three commonly used geometrical shapes—disks, rectangles, and stripes—as our choices of measurable sets $\mathcal{A}$. For the $\mathcal{A}$-distance measure, if $M = |S|$ is the total number of data points in the two collections, we show that a direct implementation of exhaustive search among the collection of all planer disks has the complexity $O(M^4)$. We present a suboptimal algorithm, the Search in sample-Centered Disks (SCD), that has the complexity $O(M^2 \log M)$. Under mild assumptions on P_i, the loss of performance of SCD diminishes as the sample size increases. For the class of axis-aligned rectangles, we show that the optimal search Search in Axis-aligned Rectangles (SAR) has complexity $O(M^3)$. A suboptimal approach Search in Diagonal-defined axis-aligned Rectangles (SDR) reduces the complexity to $O(M^2)$, again, with diminishing loss of performance under mild assumptions. For the collection of strips, we present two algorithms: Search in Axis-aligned Stripes (SAS) and Search in Random Stripes (SRS), both have complexity $O(M \log M)$. Similar analysis has also been obtained for the relative distance metric. See Table I.

We implement several algorithms and verify their performance through simulation. We also answer some practical questions arising in the implementation of the detector, *e.g.*, how to decide the detection threshold and how to estimate the minimum sample size.

B. Related Work and Organization

The problem of change detection in sensor field has been considered in different (mostly parametric) settings [4], [5]. The underlying statistical problem considered in this paper belongs to the category of two-sample nonparametric change detection. A classical approach is the Kolmogorov-Smirnov two-sample test [6] in which the empirical cumulative distributions are compared, and the maximum difference in the empirical cumulative distribution functions are used as test statistics. In a way, the methods presented in this paper generalize the idea of Kolmogorov-Smirnov test to a more general collection of measurable sets using general forms of distance measures. Indeed, the Kolmogorov-Smirnov two-sample test becomes a special case of the SAR (Search in Axis-aligned Rectangles) algorithm presented in Section IV-A.2.

There is a wealth of nonparametric change detection techniques for one-dimensional data set in which data are completely ordered. Examples include testing the number of runs (successive sample points from the same collection) such as Wald-Wolfowitz runs test, or testing the relative order of the sample points, e.g. median test, control median test, Mann-Whitney U test, and linear rank statistic tests [6]–[8]. Such techniques, however, do not have natural generalizations for the two dimensional sensor network applications.

This paper is organized as follows. Section II specifies the model and defines the detector and the estimator. Section III states the main theorems about the exponential bounds on error probabilities of the detector and the consistency of the estimator. Section IV presents the detection and estimation algorithms, and Section V provides simulation results. We conclude with comments about the strengths and weaknesses of the proposed approach.

II. The Problem Statement

A. The Model

We consider two probability measures P_1 and P_2 on the same measurable space $(X, \mathcal{F})$ where $(X, \mathcal{F}, P_i)$ models the ith random collection of the locations of the alarmed sensors[2]. Denote S_i as the set of locations of alarmed sensors in the ith collection and $S = S_1 \bigcup S_2$ the set that contains data from the two collections. We assume that, in each collection, the locations of alarmed sensors are drawn i.i.d. according to P_i[3] and the drawings in different collections are also independent. The probability measures P_i are not known, and we make no specific assumptions on their form. Note that how unalarmed sensors are distributed is not specified, we can model arbitrary correlations among them; they will not have any impact on our result. This allows us to model certain types of correlated sensor readings.

We introduce a collection $\mathcal{A} \subseteq \mathcal{F}$ of measurable sets to model the set of geographical areas that are of practical interest. The collection $\mathcal{A}$ does not have to be finite or even countable, and is part of the algorithm design. For example, we may be interested in the number of alarmed sensors in a circle centered at some location $s \in X$ with some radius r. If we define $\mathcal{A}$ as the collection of measurable subsets of X that we are interested in, it may be reasonable to focus on the probabilities of sets in $\mathcal{A}$ (rather than those in $\mathcal{F}$). The choice of $\mathcal{A}$ is subjective, of course, and it depends on the application at hand. We will focus in this paper on regular geometrical shapes: disks, rectangles, and stripes.

Given a pair of samples S_1, S_2 drawn i.i.d. from distributions P_1, P_2, and a collection $\mathcal{A} \subseteq \mathcal{F}$, we are interested in whether there is a change in probability measure on $\mathcal{A}$ and, if there is a change, where the maximum change of probability occurs. Specifically, the detection problem considered in this paper is the test of hypotheses on $\mathcal{A}$

$$\mathcal{H}_0 : P_1 = P_2 \quad \text{vs.} \quad \mathcal{H}_1 : P_1 \neq P_2$$[4]

The estimation problem, on the other hand, is to estimate the event $A^* \in \mathcal{A}$ that gives the maximum change. For example,

[2]The notation $(X, \mathcal{F}, P_i)$ is standard: X is the sample space, $\mathcal{F}$ the σ-field, P_i the probability measure.

[3]Note that the probability that an alarmed sensor reports to the fusion center may be different across sensors. This probability can be incorporated into P_i.

[4]$\mathcal{H}_0$ says $P_1(A) = P_2(A)$ for all $A \in \mathcal{A}$. $\mathcal{H}_1$ says $\exists A \in \mathcal{A}$ s.t. $P_1(A) \neq P_2(A)$.

using the $\mathcal{A}$-distance measure,

$$A^* = \arg\max_{A\in\mathcal{A}} |P_1(A) - P_2(A)|.$$

We will also consider a relative measure of change defined in Section II-B.

B. Distance Measures

To measure "change", we need some notion of distance between two probability distributions. In this paper, we will consider two distance measures: $\mathcal{A}$-distance and relative $\mathcal{A}$-distance.

$\mathcal{A}$-distance and empirical $\mathcal{A}$-distance [3] Given probability spaces $(X, \mathcal{F}, P_i)$ and a collection $\mathcal{A} \subseteq \mathcal{F}$, the *$\mathcal{A}$-distance* between P_1 and P_2 is defined as

$$d_{\mathcal{A}}(P_1, P_2) = \sup_{A\in\mathcal{A}} |P_1(A) - P_2(A)|. \tag{1}$$

The *empirical $\mathcal{A}$-distance* $d_{\mathcal{A}}(S_1, S_2)$ is similarly defined by replacing $P_i(A)$ by the empirical measure

$$S_i(A) \triangleq \frac{\triangle |S_i \bigcap A|}{|S_i|} \tag{2}$$

where $|S_i \cap A|$ is the number of points in both S_i and set A.

This notion of empirical $\mathcal{A}$-distance $d_{\mathcal{A}}(S_1, S_2)$ is related to the Kolmogorov-Smirnov two-sample statistic. For the case where the domain set is the real line, the Kolmogorov-Smirnov test considers

$$\sup_x |F_1(x) - F_2(x)|, \quad F_i(x) \triangleq P_i(\{y : y \le x\})$$

as the measure of difference between two distributions. By setting $\mathcal{A}$ to be the set of all the one-sided intervals $(-\infty, x)$, $d_{\mathcal{A}}(S_1, S_2)$ is the Kolmogorov-Smirnov statistic.

The $\mathcal{A}$-distance does not take into account the relative significance of the change. For example, one could argue that changing the probability of a set from 0.99 to 0.999 is less significant than a change from 0.001 to 0.01; the latter amounts to a ten-fold increase whereas the former represents an increase of about 1%. For applications in which small probability events are of interests, we introduce the following notion of *relative $\mathcal{A}$-distance* that takes the relative magnitudes of a change into account.

Relative and Empirical Relative $\mathcal{A}$-distance Given probability spaces $(X, \mathcal{F}, P_i)$ and a collection $\mathcal{A} \subseteq \mathcal{F}$, the *relative $\mathcal{A}$-distance* between P_1 and P_2 is defined as

$$\phi_{\mathcal{A}}(P_1, P_2) \quad = \quad \sup_{A\in\mathcal{A}} f_\phi(P_1(A), P_2(A)), \tag{3}$$

where $f_\phi : [0,1]\times[0,1] \to [0, \sqrt{2}]$ is defined as

$$f_\phi(x, y) = \begin{cases} 0 & \text{if } x = y = 0 \\ \frac{|x-y|}{\sqrt{\frac{x+y}{2}}} & \text{o.w.} \end{cases}. \tag{4}$$

The *empirical relative $\mathcal{A}$-distance* is defined similarly by replacing $P_i(A)$ with the empirical measure defined in (2).

The above definition is slightly different from that used in [3]. It is obvious that $|P_1(A) - P_2(A)|$ is a metric. The proof that $\frac{|P_1(A)-P_2(A)|}{\sqrt{\frac{P_1(A)+P_2(A)}{2}}}$ is a metric follows from [9]. Note that in general $d_{\mathcal{A}}(P_1, P_2) = 0$ or $\phi_{\mathcal{A}}(P_1, P_2) = 0$ does not imply $P_1 = P_2$, but implies $P_1(A) = P_2(A), \forall A \in \mathcal{A}$. If we only care about sets in $\mathcal{A}$, $d_{\mathcal{A}}$ and $\phi_{\mathcal{A}}$ defined above are pseudo-metrics on $\mathcal{A}$.

C. Detection and Estimation

With the distance measure defined, we can now specify the class of detectors and estimators considered in this paper.

Detector $\delta(S_1, S_2; \epsilon)$: Given two collections of sample points S_1 and S_2, drawn i.i.d from probability distributions P_1 and P_2 respectively, and threshold $\epsilon \in (0, 1)$, for hypotheses $\mathcal{H}_0$ vs. $\mathcal{H}_1$, the detector[5] using the $\mathcal{A}$-distance is defined as

$$\delta_{d_{\mathcal{A}}}(S_1, S_2; \epsilon) = \begin{cases} 1 & \text{if } d_{\mathcal{A}}(S_1, S_2) > \epsilon \\ 0 & \text{otherwise} \end{cases} \tag{5}$$

The detector $\delta_{\phi_{\mathcal{A}}}(S_1, S_2; \epsilon)$ using the relative $\mathcal{A}$-distance is defined the same way by replacing $d_{\mathcal{A}}(S_1, S_2)$ by $\phi_{\mathcal{A}}(S_1, S_2)$ and letting $\epsilon \in (0, \sqrt{2})$.

Assuming that a change of probability distribution has occurred, we define the estimator for the event that gives the maximum change in probabilities.

Estimator $\hat{A}^*(S_1, S_2)$: Given two collection of sample points S_1 and S_2, drawn i.i.d from probability distributions P_1 and P_2 respectively, the estimator for the event that gives the maximum change of probability is defined as

$$\hat{A}^*_{d_{\mathcal{A}}}(S_1, S_2) = \arg\max_{A\in\mathcal{A}} |S_1(A) - S_2(A)|.$$

The estimator $\hat{A}^*_{\phi_{\mathcal{A}}}(S_1, S_2)$ using the relative $\mathcal{A}$-distance is defined similarly.

The definitions given above require searches in a possibly infinite collection of sets. At the moment, we only specify what the outcome should be without addressing the algorithmic procedure generating it. We will address that issue in Section IV.

III. Performance Guarantee

We present in this section consistency results for the detector and estimator presented earlier. The results are given in the forms of error exponents.

First let us look at some technical preliminary from [2]. For measurable space $(X, \mathcal{F})$, let $\mathcal{A} \subseteq \mathcal{F}$. We say a set $S \subset X$ is *shatterable* by $\mathcal{A}$ if for all $B \subseteq S$, $\exists A \in \mathcal{A}$ s.t.

$$B = A \cap S.$$

VC-Dimension The Vapnik-Chervonenkis dimension of a collection $\mathcal{A}$ of sets is

$$\text{VC-d}(\mathcal{A}) = \sup\{n : \exists B \text{ s.t. } |B| = n \text{ and } B \text{ is shatterable by } \mathcal{A}\}.$$

The VC-dimension of a class of sets quantifies its ability to separate sets of points. Intuitively the VC-dimension of a class $\mathcal{A}$ is the maximum number of free parameters needed

[5]We use the convention that the detector gives the value 1 for $\mathcal{H}_1$ and 0 for $\mathcal{H}_0$.

to specify a set in $\mathcal{A}$. For example, if $\mathcal{A} = \{\text{2D disks}\}$, then we see that at most 3 free parameters are needed — x, y-coordinates of the center and a radius, and it is shown that the VC-dimension of $\mathcal{A}$ is indeed 3 [10].

Note that the VC-dimension of a class may be infinite,*e.g.*, VC-dimension of the entire σ-field $\mathcal{F}$ is ∞ because any set is shatterable by $\mathcal{F}$.

Theorem 3.1 (Detector Error Exponents): Given probability spaces $(X, \mathcal{F}, P_i)$ and a collection $\mathcal{A} \subseteq \mathcal{F}$ with finite VC-dimension d, let $S_i \subset X$ be a set of n sample points drawn according to P_i. The false alarm probabilities for the detectors defined in (5) are bounded by

$$P_F(\delta_{d_\mathcal{A}}) \leq 8(2n+1)^d e^{-n\epsilon^2/32}, \quad (6)$$

$$P_F(\delta_{\phi_\mathcal{A}}) \leq 2(2n+1)^d e^{-n\epsilon^2/4}. \quad (7)$$

Furthermore, if $d_\mathcal{A}(P_1, P_2) > \epsilon$ and $\phi_\mathcal{A}(P_1, P_2) > \epsilon$, the miss detection probabilities satisfy, respectively,

$$P_M(\delta_{d_\mathcal{A}}, P_1, P_2) \leq 8(2n+1)^d e^{-n[d_\mathcal{A}(P_1,P_2)-\epsilon]^2/32}, \quad (8)$$

$$P_M(\delta_{\phi_\mathcal{A}}, P_1, P_2) \leq 16(2n+1)^d e^{-n[\phi_\mathcal{A}(P_1,P_2)-\epsilon]^2/16}. \quad (9)$$

Proof: See Appendix.

A few remarks are in order. First, if the maximum change between P_1 and P_2 on $\mathcal{A}$ exceeds ϵ, the detector detects the change with probability arbitrarily close to 1 as the sample size goes to infinity. Similarly, if there is no change in P_i on $\mathcal{A}$, then the probability of false alarm also goes to zero. Notice that the decay rates of the error probabilities are different when the two different distance measures are used; from (6,7), the decay rate of false alarm probabilities for the detector using $\phi_\mathcal{A}$ is eight times that using $d_\mathcal{A}$.

Second, the above theorem provides a way of deciding the detection threshold ϵ for a particular detection criterion. For example, the threshold (not necessarily optimal) of the Neyman-Pearson detection for a given size α can be obtained from the bounds on false alarm probabilities. Theorem 3.1 suggests that we should choose (n, ϵ) such that

$$8(2n+1)^d e^{-n\epsilon^2/32} \leq \alpha \quad \text{for } \delta_{d_\mathcal{A}} \quad (10)$$

$$2(2n+1)^d e^{-n\epsilon^2/4} \leq \alpha \quad \text{for } \delta_{\phi_\mathcal{A}}. \quad (11)$$

Taking $\epsilon(n)$ to make the inequalities equal gives a threshold

$$\epsilon(n) = \begin{cases} \sqrt{\frac{32}{n}\log\frac{8(2n+1)^d}{\alpha}} & \text{for } \delta_{d_\mathcal{A}} \\ \sqrt{\frac{4}{n}\log\frac{2(2n+1)^d}{\alpha}} & \text{for } \delta_{\phi_\mathcal{A}} \end{cases} \quad (12)$$

We shall think of $\epsilon(n)$ as a measure of detector sensitivity. From (8,9) in Theorem 3.1, we see that miss detection probability starts to drop exponentially when $\epsilon(n) < d_\mathcal{A}(P_1, P_2)$ or $\epsilon(n) < \phi_\mathcal{A}(P_1, P_2)$. Thus, roughly, $\epsilon(n)$ is a lower bound on the amount of changes in order for the change to be detected with high probability. Furthermore, the smaller the $\epsilon(n)$, the larger the values of $[d_\mathcal{A}(P_1, P_2) - \epsilon(n)]^2/32$ and $[\phi_\mathcal{A}(P_1, P_2) - \epsilon(n)]^2/16$, and the lower the upper bound on miss detection probability.

Third, note that the VC-dimension d of $\mathcal{A}$ has diminishing effects on the rate of decay of error probabilities. The selection of $\mathcal{A}$, however, may affect the error exponent through $d_\mathcal{A}$ or $\phi_\mathcal{A}$. Furthermore, the selection of $\mathcal{A}$ has a significant impact on the complexity of practically implementable algorithms.

Finally, we should also note that, while we have stated the above theorem under $|S_i| = n$, the results generalized easily to the case when two collections have difference sizes but they are proportional.

The consistency of the estimator is implied by the following theorem.

Theorem 3.2: Given probability spaces $(X, \mathcal{F}, P_i)$ and a collection $\mathcal{A} \subseteq \mathcal{F}$ with finite VC-dimension, if $\arg\max_{A\in\mathcal{A}} |P_1(A) - P_2(A)|$ is well defined, *i.e.*, it is unique, then with probability going to 1 as $n \to \infty$ (with high probability),

$$\arg\max_{B\in\mathcal{A}} |S_1(B) - S_2(B)| = \arg\max_{A\in\mathcal{A}} |P_1(A) - P_2(A)|.$$

Similarly, if $\arg\max_{A\in\mathcal{A}} f_\phi(P_1(A), P_2(A))$ is well defined, then with high probability

$$\arg\max_{B\in\mathcal{A}} f_\phi(S_1(B), S_2(B)) = \arg\max_{A\in\mathcal{A}} f_\phi(P_1(A), P_2(A)).$$

Proof: See Appendix.

IV. Algorithms

We now turn our attention to practically implementable algorithms and their complexities. The key step is to obtain test statistics within a finite number of operations, preferably with the complexity that scales well with the total number of data points $M = |S_1 \bigcup S_2|$.

Given sample points $S = S_1 \bigcup S_2$ and a possibly infinite collection of sets $\mathcal{A}$, we need to reduce the search in $\mathcal{A}$ to a search in a *finite* collection $\mathcal{H}(S) \subset \mathcal{A}$, and replace $d_\mathcal{A}(S_1, S_2)$ by $d_\mathcal{H}(S_1, S_2)$. If $\mathcal{H}$ is not chosen properly, such a reduction of the search domain may lead to a loss of performance. Thus we need the notion of completeness when choosing the search domain.

Completeness Given $\mathcal{A}$ being a collection of measurable subsets of space X, and $S \subset X$ be a set of points in X. Let $\mathcal{H}(S) \subset \mathcal{A}$ be a finite sub-collection of measurable sets which is a function of S. We call the collection $\mathcal{H}(S)$ *complete for S with respect to $\mathcal{A}$* if $\forall A \in \mathcal{A}$, there exists a $B \in \mathcal{H}(S)$ such that $S \cap A = S \cap B$.

The significance of the completeness is that, if $\mathcal{H}(S_1 \cup S_2)$ is complete w.r.t. $\mathcal{A}$, then $d_\mathcal{A}(S_1, S_2) = d_\mathcal{H}(S_1, S_2)$ and $\phi_\mathcal{A}(S_1, S_2) = \phi_\mathcal{H}(S_1, S_2)$.

For the choice of $\mathcal{A}$, we consider regular geometric areas, *e.g.*, disks, rectangles, and stripes. We present next six algorithms for different choices of $\mathcal{A}$ and sub-collection $\mathcal{H}$. We first present complete algorithms, i.e. the sub-collection $\mathcal{H}$ is complete with respect to $\mathcal{A}$. Next we give a couple of heuristic algorithms which simplify the computation at the cost of a loss in completeness.

Hereinafter all sets defined are closed sets unless otherwise stated.

A. Complete Algorithms

1) Search in Planar Disks (SPD): Let $\mathcal{A}$ be the collection of two dimensional disks. Let VC-d denote the VC-dimension of a class. The following result is proved by [10]:

Proposition 4.1:

$$\text{VC-d}(\mathcal{A}) = 3.$$

For the set of sample points $S \subseteq X$, consider the finite sub-collection of $\mathcal{A}$ defined by

$$\mathcal{H}_{\text{D}}(S) \stackrel{\Delta}{=} \bigcup_{(s_i,s_j,s_k)\in\mathcal{T}} \mathcal{H}_{\text{D}}(s_i, s_j, s_k) \tag{13}$$

where

$$\mathcal{T} \stackrel{\Delta}{=} \{s_i, s_j, s_k \in S^3 : s_i, s_j, s_k \text{ are not collinear}\},$$

and

$$\mathcal{H}_{\text{D}}(s_i, s_j, s_k) \stackrel{\Delta}{=} \{D(s_i, s_j, s_k), D(s_i, s_j, s_k) \setminus \{s_i\},$$
$$D(s_i, s_j, s_k) \setminus \{s_j\}, \ldots, D(s_i, s_j, s_k) \setminus \{s_i, s_j, s_k\}\}$$

where $D(s_i, s_j, s_k)$ is the disk with s_i, s_j, and s_k on its boundary, *i.e.*, $\mathcal{H}_{\text{D}}(s_i, s_j, s_k)$ is $D(s_i, s_j, s_k)$ and all the 7 variations for excluding some of the 3 boundary points. See Figure 2.

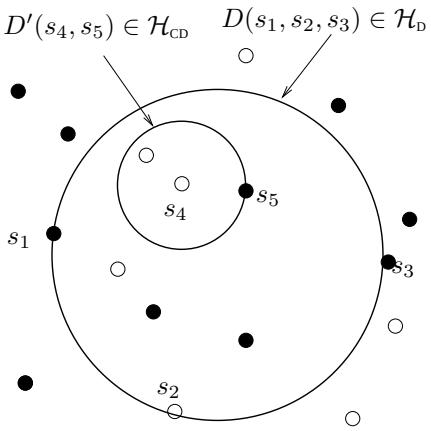


Fig. 2. Members of $\mathcal{H}_{\text{D}}$ and $\mathcal{H}_{\text{CD}}$; ○: sample point in S_1, •: sample point in S_2

In [11] we have proved the following result:

Proposition 4.2: Let $\mathcal{A}$ be the collection of two dimensional disks. For S_1 and S_2 drawn from P_1 and P_2, if P_1 and P_2 are such that any set with Lebesgue measure 0 has probability 0[6], then the finite collection $\mathcal{H}_{\text{D}}(S_1 \cup S_2)$ in (13) is complete with respect to $\mathcal{A}$ a.s.(almost surely).

With $\mathcal{H}_{\text{D}}(S)$ defined above, the algorithm SPD($d_{\mathcal{A}}$)—Search in Planar Disks using distance metric $d_{\mathcal{A}}$—is given by

$$\max_{A\in\mathcal{H}_{\text{D}}} |S_1(A) - S_2(A)|.$$

Algorithm SPD($d_{\mathcal{A}}$) includes three steps: (i) generating elements of $\mathcal{H}_{\text{D}}$; (ii) computing $\left|\frac{|S_1\cap A|}{|S_1|} - \frac{|S_2\cap A|}{|S_2|}\right|$ by counting $|S_1 \cap A|$ and $|S_2 \cap A|$ for every $A \in \mathcal{H}_{\text{D}}$, and (iii) finding the maximum.

Algorithm SPD($\phi_{\mathcal{A}}$) (Search in Planar Disks using the metric $\phi_{\mathcal{A}}$) is the same as SPD($d_{\mathcal{A}}$) except in step (ii) where the relative empirical measure is computed.

We now analyze the complexity of SPD. The complexities of both SPD($d_{\mathcal{A}}$) and SPD($\phi_{\mathcal{A}}$) are $O(M^4)$ for sample size $M = |S_1 \cup S_2|$. This is because there are $O(M^3)$ disks to consider, and the counting of $|S_1 \cap A|$ and $|S_2 \cap A|$ for each disk takes M steps.

2) Search in Axis-aligned Rectangles (SAR): We now consider the collection $\mathcal{A}$ of axis-aligned rectangles. Then we have the following property:

Proposition 4.3:

$$\text{VC-d}(\mathcal{A}) = 4.$$

Proof: It is easy to see that VC-d($\mathcal{A}$) ≥ 4. See Fig. 3. The set

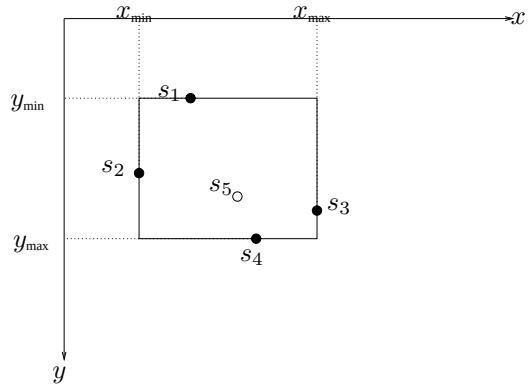


Fig. 3.

$\{s_1, s_2, s_3, s_4\}$ is shatterable by $\mathcal{A}$.

For any set S of more than 4 points. Let $x_{\min}$, $x_{\max}$, $y_{\min}$, $y_{\max}$ be the minimum and the maximum x, y-coordinates for points in S, and let the points with these coordinates be s_1, s_2, s_3, s_4 (some of them can be the same). Then any axis-aligned rectangle containing $\{s_1,\ s_2,\ s_3,\ s_4\}$ contains S. The subset $\{s_1,\ s_2,\ s_3,\ s_4\}$ cannot be obtained by shattering S with $\mathcal{A}$, and S is not shatterable. Hence VC-d($\mathcal{A}$) ≤ 4. ■

Given samples S_1 and S_2, let $S = S_1 \cup S_2 = \{(x_1, y_1), \cdots, (x_M, y_M)\}$ where, at the cost of $O(M \log M)$, we may assume that $x_1 \leq x_2 \leq \cdots \leq x_M$. Let the finite collection $\mathcal{H}_{\text{R}}(S)$ be defined by

$$\mathcal{H}_{\text{R}}(S) \stackrel{\Delta}{=} \{R(y_i, y_j, x_m, x_n) : (x_k, y_k) \in S, k = i, j, m, n\} \tag{14}$$

where $R(y_i, y_j, x_m, x_n)$ is the rectangle defined by the four lines $y = y_i, y = y_j, x = x_m, x = x_n$. See Figure 4.

Proposition 4.4: Let $\mathcal{A}$ be the class of two dimensional axis-aligned rectangles. Given S_1 and S_2, the finite collection $\mathcal{H}_{\text{R}}(S_1 \cup S_2)$ in (14) is complete with respect to $\mathcal{A}$.

The reason for this proposition is that for any axis-aligned rectangle R and given S, we can find axis-aligned rectangle

[6]This is true if P_1,P_2 are absolutely continuous, *i.e.*, having pdf, because any measurable function has integration 0 on a 0-measure set.

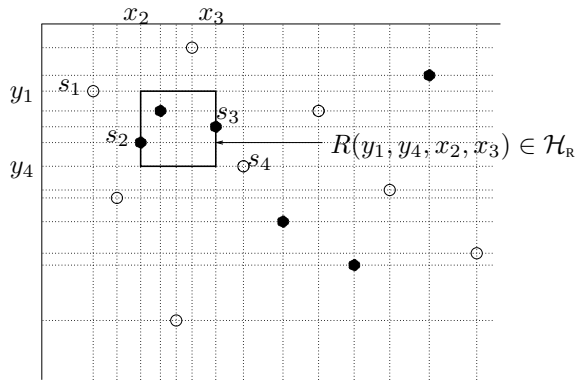


Fig. 4. Members of $\mathcal{H}_{\text{R}}$

R' such that $R' \cap S = R \cap S$ and R' has at least one sample point on each side of the boundary, where points on different sides are not necessarily different. Since $\mathcal{H}_{\text{R}}$ includes all those rectangles, it is complete w.r.t. $\mathcal{A}$.

Algorithm SAR($d_{\mathcal{A}}$) computes $d_{\mathcal{H}_{\text{R}}}(S_1, S_2)$. Because of the ordering in x_i's, the collection $\mathcal{H}_{\text{R}}$ allows a recursive calculation of distance measures. Specifically, for fixed y_i and y_j s.t. $y_i \le y_j$, define

$$f_{ij}^k(n) \triangleq |S_k \cap R(y_i, y_j, x_1, x_n)|/|S_k|, k = 1, 2 \quad (15)$$
$$F_{ij}(n) = f_{ij}^1(n) - f_{ij}^2(n) \quad (16)$$

Then $f_{ij}^k(n)$ ($n = 1, \ldots, M$) can be computed recursively by

$$f_{ij}^k(n) = \begin{cases} f_{ij}^k(n-1) + \frac{1}{|S_k|} & y_n \in [y_i, y_j], (x_n, y_n) \in S_k \\ f_{ij}^k(n-1) & \text{otherwise} \end{cases}.$$

Then find

$$i_{\max} = \arg\max_n F_{ij}(n), \quad i_{\min} = \arg\min_n F_{ij}(n)$$
$$l \triangleq \min\{i_{\max}, i_{\min}\} + 1, \quad u \triangleq \max\{i_{\max}, i_{\min}\}$$

The optimal rectangle, for fixed y_i and y_j, is then given by $R(y_i, y_j, x_l, x_u)$, and the maximum difference in empirical probabilities is given by $F_{ij}(i_{\max}) - F_{ij}(i_{\min})$.

Finally, compute

$$d_{\mathcal{H}_{\text{R}}}(S_1, S_2) = \max_{i,j: y_i \le y_j} (F_{ij}(i_{\max}) - F_{ij}(i_{\min})).$$

The pair (i, j) that achieves this maximum gives the best rectangle in $\mathcal{H}_{\text{R}}$.

Algorithm SAR($\phi_{\mathcal{A}}$) computes $\phi_{\mathcal{H}_{\text{R}}}(S_1, S_2)$. For fixed y_i and y_j ($y_i \le y_j$), we compute $f_{ij}^1(n)$ and $f_{ij}^2(n)$ for $n = 1, \ldots, M$ as before. Compute empirical probabilities for every pair $x_m < x_n$ by

$$S_k(R(y_i, y_j, x_m, x_n)) = f_{ij}^k(n) - f_{ij}^k(m), \ k = 1, 2 \quad (17)$$

Then optimizing over all the pairs of x's and y's

$$\max_{\substack{i,j,m,n: \\ y_i \le y_j, m < n}} \frac{|S_1(R(y_i, y_j, x_m, x_n)) - S_2(R(y_i, y_j, x_m, x_n))|}{\sqrt{\frac{S_1(R(y_i, y_j, x_m, x_n)) + S_2(R(y_i, y_j, x_m, x_n))}{2}}}$$

gives $\phi_{\mathcal{H}_{\text{R}}}(S_1, S_2)$ and the best rectangle.

We now analyze the complexity of Algorithm SAR. SAR($d_{\mathcal{A}}$) has complexity $O(M^3)$, and SAR($\phi_{\mathcal{A}}$) has complexity $O(M^4)$. This is because in computing $d_{\mathcal{A}}$ we can use the fact that

$$\max_{m,n} |(f_{ij}^1(n) - f_{ij}^1(m)) - (f_{ij}^2(n) - f_{ij}^2(m))|$$
$$= \max_{m,n} |(f_{ij}^1(n) - f_{ij}^2(n)) - (f_{ij}^1(m) - f_{ij}^2(m))| \quad (18)$$
$$= \max_n (f_{ij}^1(n) - f_{ij}^2(n)) - \min_m (f_{ij}^1(m) - f_{ij}^2(m)) \quad (19)$$

and reduce the two-variable optimization to two one-variable optimizations, which are done in linear time. To compute $\phi_{\mathcal{A}}$, however, we have to check all the $O(M^2)$ (x_m, x_n) pairs. The search is then repeated for all the $O(M^2)$ (y_i, y_j) pairs. Note that the VC-dimension of the collection of axis-aligned rectangles is 4 while the VC dimension of the collection of planar disks is 3, which results in a larger sample size M for Algorithm SAR as we discuss later.

3) Search in Axis-aligned Stripes (SAS): The complexities of algorithms SPD and SAR may be formidable for large M. This urgent need of reducing complexity gives birth to a simplified algorithm that deals with axis-aligned stripes. The basic idea is to project sample points onto x and y coordinates, and then perform change detection/estimation on each coordinate.

Let $\mathcal{A}$ be the collection of vertical stripes, i.e., axis-aligned rectangles with height equal to the field height. Similarly, let $\mathcal{B}$ be the collection of horizonal stripes. The following property is true:

Proposition 4.5:

$$\text{VC-d}(\mathcal{A} \cup \mathcal{B}) = 4.$$

Proof: It is easy to see that VC-d($\mathcal{A} \cup \mathcal{B}$) ≥ 4. See Fig. 5.

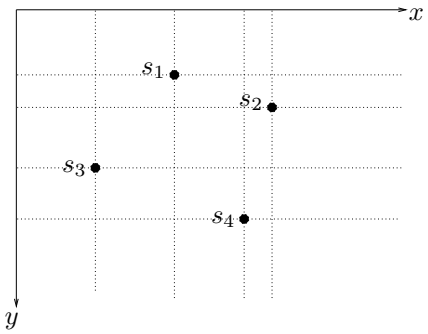


Fig. 5.

The set $\{s_1, s_2, s_3, s_4\}$ is shatterable by $\mathcal{A} \cup \mathcal{B}$.

For any set S of more than 4 points. Let s_{l}, s_{r}, s_{u}, s_{o} be the points with the minimum and the maximum x, y-coordinates in S accordingly (not necessarily different). Then any vertical stripe containing $\{s_{\text{l}}, s_{\text{r}}\}$ contains S, and any horizonal stripe containing $\{s_{\text{u}}, s_{\text{o}}\}$ also contains S. The subset $\{s_{\text{l}}, s_{\text{r}}, s_{\text{u}}, s_{\text{o}}\}$ cannot be obtained by shattering S with $\mathcal{A} \cup \mathcal{B}$, and thus S is not shatterable by $\mathcal{A} \cup \mathcal{B}$. Hence VC-d($\mathcal{A} \cup \mathcal{B}$) ≤ 4.

■

Given a collection of sample points $S = S_1 \cup S_2$, consider finite subsets $\mathcal{H}_{\text{V}}(S) \subset \mathcal{A}$ and $\mathcal{H}_{\text{H}}(S) \subset \mathcal{B}$ defined by

$$\mathcal{H}_{\text{V}}(S) \triangleq \{V(x_i, x_j) : s_i = (x_i, y_i), s_j = (x_j, y_j) \in S\} \quad (20)$$

$$\mathcal{H}_{\text{H}}(S) \triangleq \{H(y_k, y_l) : s_k = (x_k, y_k), s_l = (x_l, y_l) \in S\} \quad (21)$$

where $V(x_i, x_j)$ is the vertical stripe with left and right boundary x_i and x_j, and $H(y_k, y_l)$ is the horizonal stripe with lower and upper boundary y_k and y_l. See Figure 6.

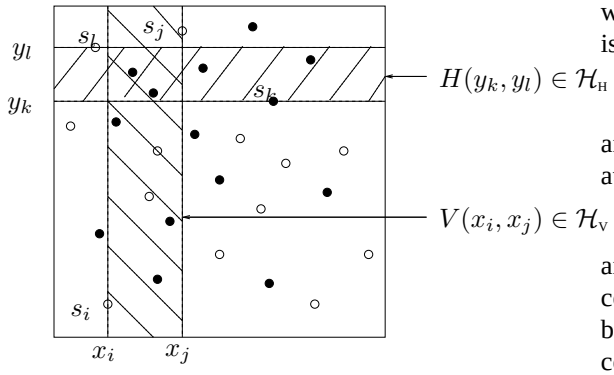


Fig. 6. Members of $\mathcal{H}_{\text{V}}$ and $\mathcal{H}_{\text{H}}$

Proposition 4.6: Let $\mathcal{A}$ be the class of vertical stripes and $\mathcal{B}$ be the class of horizonal stripes. Given S_1 and S_2, the finite collection $\mathcal{H}_{\text{V}}(S_1 \cup S_2) \cup \mathcal{H}_{\text{H}}(S_1 \cup S_2)$ defined in (20) and (21) is complete with respect to $\mathcal{A} \cup \mathcal{B}$.

The proposition is easy to verify because for any axis-aligned stripe, we can find another axis-aligned stripe with the same intersection with S and at least one sample point on each boundary. Thus it suffices to consider stripes with sample points on the boundary.

Given S, Algorithm SAS($d_{\mathcal{A}}$) performs the following search

$$\max_{A \in \mathcal{H}_{\text{V}} \cup \mathcal{H}_{\text{H}}} |S_1(A) - S_2(A)|.$$

The algorithm includes the following steps: (i) project sample points onto x and y coordinates; (ii) sort the projected sample points into increasing order; (iii) in the x coordinate (we have $x_1 \le x_2 \le \cdots \le x_M$), for $i = 1, \ldots, M$, compute $f_x^k(i) \triangleq S_k(V(0, x_i))$ ($k = 1, 2$) recursively by

$$f_x^k(i) = \begin{cases} f_x^k(i-1) + \frac{1}{|S_k|} & \text{if } s_i \in S_k \\ f_x^k(i-1) & \text{otherwise} \end{cases}, \quad (22)$$

and then compute $F_x(i) \triangleq f_x^1(i) - f_x^2(i)$; compute $F_y(j) \triangleq S_1(H(0, y_j)) - S_2(H(0, y_j))$ similarly; (iv) find

$$m_1 = \arg\max_i F_x(i), \ m_2 = \arg\min_i F_x(i).$$

$$n_1 = \arg\max_j F_y(j), \ n_2 = \arg\min_j F_y(j).$$

We then have

$$\max_{A \in \mathcal{H}_{\text{V}} \cup \mathcal{H}_{\text{H}}} |S_1(A) - S_2(A)| = \max(F_x(m_1) - F_x(m_2), F_y(n_1) - F_y(n_2)) \quad (23)$$

and the estimation of the changed area is $V(x_{m_1}, x_{m_2})$ if $F_x(m_1) - F_x(m_2) > F_y(n_1) - F_y(n_2)$, or $H(y_{n_1}, y_{n_2})$ otherwise.

Algorithm SAS($\phi_{\mathcal{A}}$) does the same in steps (i),(ii) and (iii), but (iv) is changed to finding

$$\phi_{\mathcal{H}_{\text{V}}}(S_1, S_2) = \max_{i,j: i<j} \frac{|S_1(V(x_i, x_j)) - S_2(V(x_i, x_j))|}{\sqrt{\frac{S_1(V(x_i,x_j)) + S_2(V(x_i,x_j))}{2}}} \quad (24)$$

where $S_k(V(x_i, x_j))$ is given by $f_x^k(j) - f_x^k(i)$. $\phi_{\mathcal{H}_{\text{H}}}(S_1, S_2)$ is computed similarly. Then

$$\phi_{\mathcal{H}_{\text{V}} \cup \mathcal{H}_{\text{H}}}(S_1, S_2) = \max(\phi_{\mathcal{H}_{\text{V}}}(S_1, S_2), \phi_{\mathcal{H}_{\text{H}}}(S_1, S_2))$$

and the changed area is the stripe on which the maximum is attained.

Now we analyze the complexities of Algorithm SAS($d_{\mathcal{A}}$) and Algorithm SAS($\phi_{\mathcal{A}}$). Given $M = |S_1 \cup S_2|$, the complexity of Algorithm SAS($d_{\mathcal{A}}$) is $O(M \log M)$. This is because by projection, we only need to perform two linear-complexity searches. Now the dominating part is the sorting of sample points, which takes $O(M \log M)$. The complexity of Algorithm SAS($\phi_{\mathcal{A}}$) is $O(M^2)$ because in the two-variable optimization there are $O(M^2)$ (x_i, x_j) pairs to consider.

4) Search in Random Stripes (SRS): Note that in Algorithm SAS the choice of x and y axes for projection is subjective, and this choice should be part of algorithm design. When we know nothing about the change, introducing randomness may give more robustness to the algorithms.

For θ randomly selected from $[0, \frac{\pi}{2}]$, chose $\mathcal{A}^\theta$ to be the collection of vertical stripes rotated (counter-clockwise) by θ, and $\mathcal{B}^\theta$ to be the collection of horizonal stripes rotated by θ. Define $\mathcal{H}_{\text{V}}^\theta(S)$ and $\mathcal{H}_{\text{H}}^\theta(S)$ to be members of $\mathcal{A}^\theta$, $\mathcal{B}^\theta$ accordingly, with sample points on the boundary, which is similar to definitions (20,21).

We claim similar properties for $\mathcal{A}^\theta \cup \mathcal{B}^\theta$ and $\mathcal{H}_{\text{V}}^\theta(S) \cup \mathcal{H}_{\text{H}}^\theta(S)$, *i.e.*, VC-d($\mathcal{A}^\theta \cup \mathcal{B}^\theta$) $= 4$ and $\mathcal{H}_{\text{V}}^\theta(S) \cup \mathcal{H}_{\text{H}}^\theta(S)$ is complete with respect to $\mathcal{A}^\theta \cup \mathcal{B}^\theta$. Note that introducing θ does not increase the VC-dimension to 5 because the projection direction is randomly chosen but not optimized over.

Algorithm SRS is a randomized variation of Algorithm SAS. It is based on the same projection and search idea as in Algorithm SAS. The difference is when performing the projection, we project sample points onto random directions instead of the fixed directions of x and y axes. The rest of the algorithm is the same as Algorithm SAS.

Algorithm SRS has the same order of complexity as Algorithm SAS in computing both $d_{\mathcal{A}}$ and $\phi_{\mathcal{A}}$. The advantage of Algorithm SRS is that it is more robust than Algorithm SAS. Specifically, as a randomized algorithm, SRS will perform equally well under a wider range of change patterns (the way

change occurs) while SAS can be affected significantly by the change pattern. For example, SAS is vulnerable to the pattern where changes always occur along a tilted line of angle 45° or 135°, because in that case the increasing and decreasing parts of the change will largely get cancelled when projected onto axes.

A quick comment is in order. Both Algorithm SAS and Algorithm SRS can be easily generalized to algorithms of multiple projections. By doing multiple projections and line searches, we can increase the accuracy of the algorithm at the cost of a constant factor increase in the complexity.

B. Heuristic Algorithms

Some complete algorithms may be good in performance but too expensive to implement in practice, while the simplified complete algorithms SAS and SRS may be not sensitive enough to detect the changes despite their improved complexities. A trade-off is heuristic algorithms which have lower complexities than their complete counterparts and perform reasonably well for certain classes of distributions.

1) Search in sample-Centered Disks (SCD): In calculating the distances on $\mathcal{H}_{\text{D}}$ in SPD, it is difficult to reuse the calculation since sample-defined disks may overlap in arbitrary ways. We define here a different sub-collection in which disks form nested sets, which allows the recursive computation of distances.

Let $\mathcal{A}$ be the collection of two dimensional disks. Given sample $S = S_1 \bigcup S_2$, $\mathcal{H}_{\text{CD}}(S) \subset \mathcal{A}$ is the sub-collection of sample-centered disks defined by

$$\mathcal{H}_{\text{CD}}(S) \triangleq \{D'(s_i, s_j) : s_i, s_j \in S\} \tag{25}$$

where $D'(s_i, s_j)$ is the disk with s_i at the center and s_j on the boundary. See Figure 2.

Proposition 4.7:

$$\text{VC-d}(\mathcal{H}_{\text{CD}}) = 2.$$

Proof:

It is easy to see that VC-d$(\mathcal{H}_{\text{CD}}) \geq 2$ because any set of two points can be shattered (a singleton also belongs to $\mathcal{H}_{\text{CD}}$).

For any set S of 3 points, *i.e.,* $S = \{s_1,\ s_2,\ s_3\}$. Let

$$|s_1 s_2| = \max_{i,\, j \in \{1,\, 2,\, 3\}} |s_i s_j|.$$

Then $\{s_1,\ s_2\}$ cannot be shattered (*i.e.,* obtained by shattering) because the only way to shatter it is by $D'(s_1, s_2)$ or $D'(s_2, s_1)$, but they both contain s_3. Hence any such S is not shatterable, and VC-d$(\mathcal{H}_{\text{CD}}) \leq 2$.

■

Unfortunately, $\mathcal{H}_{\text{CD}}$ is not complete with respect to $\mathcal{A}$. For some classes of probability distributions, however, it turns out that SCD has the same performance as SPD asymptotically. For example, if there exists some center point such that any neighborhood around the center has reasonably high probability, SCD is expected to perform almost as well as SPD. Generally, if probability measures P_i are such that any disk with positive Lebesgue measure has positive probability, then the loss of performance vanishes asymptotically. Consider a disk and an arbitrary neighborhood of its center, the strong law of large numbers guarantees that as sample size goes to infinity, there is a sample point within this neighborhood of the center almost surely. This implies that as sample size goes to infinity, Algorithm SCD will give the same output as Algorithm SPD, *i.e.,* the search of SCD is asymptotically complete.

Algorithm SCD$(d_{\mathcal{A}})$ computes

$$\max_{A \in \mathcal{H}_{\text{CD}}} |S_1(A) - S_2(A)|.$$

The presence of increasing subsets allows the counting procedure to be incremental, i.e. fix a center and count the number of sample points recursively from the inner-most disk to the outer-most disk.

Algorithm SCD$(d_{\mathcal{A}})$ does the following:

Fix a center s_i and define

$$F_i(j) \quad \triangleq \quad S_1(D'(s_i, s_j)) - S_2(D'(s_i, s_j)) \tag{26}$$

where $S_k(D'(s_i, s_j)), k \in \{1, 2\}$ is the empirical probability of $D'(s_i, s_j)$ in S_k. First sort the sample points into increasing order $s_{j_1}, s_{j_2}, \ldots$ according to their distance to s_i[7] ($s_{j_1} = s_i$), and then set $F_i(j_0) = 0$ and compute $F_i(j_k)$ ($k = 1, 2, \ldots, M$) recursively by

$$F_i(j_k) = \begin{cases} F_i(j_{k-1}) + \frac{1}{|S_1|} & \text{if } s_{j_k} \in S_1 \\ F_i(j_{k-1}) - \frac{1}{|S_2|} & \text{if } s_{j_k} \in S_2 \end{cases}.$$

Next compute

$$j^*(i) \quad = \quad \arg\max_j |F_i(j)|. \tag{27}$$

The search is repeated for all possible s_i. Finally, we find the maximum among $|F_i(j^*(i))|, \forall i$, i.e.

$$i_{\max} \quad = \quad \arg\max_i |F_i(j^*(i))|. \tag{28}$$

Then the optimal disk in $\mathcal{H}_{\text{CD}}$ for $\mathcal{A}$-distance is given by $D'(s_{i_{\max}}, s_{j^*(i_{\max})})$, and the maximum difference is

$$\max_{A \in \mathcal{H}_{\text{CD}}} |S_1(A) - S_2(A)| = |F_{i_{\max}}(j^*(i_{\max}))|.$$

Algorithm SCD$(\phi_{\mathcal{A}})$ computes

$$\max_{A \in \mathcal{H}_{\text{CD}}} \frac{|S_1(A) - S_2(A)|}{\sqrt{\frac{S_1(A) + S_2(A)}{2}}}.$$

Clearly when computing $F_i(j)$, we can get $S_1(D'(s_i, s_j))$ and $S_2(D'(s_i, s_j))$ by similar update, so we can compute

$$G_i(j) = \frac{|S_1(D'(s_i, s_j)) - S_2(D'(s_i, s_j))|}{\sqrt{\frac{S_1(D'(s_i, s_j)) + S_2(D'(s_i, s_j))}{2}}}.$$

Then

$$\max_{A \in \mathcal{H}_{\text{CD}}} \frac{|S_1(A) - S_2(A)|}{\sqrt{\frac{S_1(A) + S_2(A)}{2}}} = \max_{i,j} G_i(j).$$

[7]This sort is at the cost of $O(M \log M)$.

The complexities of Algorithm SCD($d_{\mathcal{A}}$) and Algorithm SCD($\phi_{\mathcal{A}}$) are of the same order. Their complexity, compared with the $O(M^4)$ complexity of Algorithm SPD, is reduced to $O(M^2 \log M)$. The dominating term is the sorting of the sample points according to their distances to a certain sample point, which takes $O(M \log M)$ for each center, and is repeated for M centers.

2) Search in Diagonal-defined axis-aligned Rectangles (SDR): Algorithm SDR is a heuristic simplification of Algorithm SAR. A major drawback of Algorithm SAR is that it is much slower in computing $\phi_{\mathcal{A}}$ distance ($O(M^4)$ compared to $O(M^3)$ in computing $d_{\mathcal{A}}$). Aiming at reducing the cost of computing $\phi_{\mathcal{A}}$ for rectangles, we propose a simplified variation of SAR: Algorithm SDR. Inspired by Kolmogorov-Smirnov two-sample test [6], we reduce the search to the class of axis-aligned rectangles having sample points on diagonal vertices.

Let $\mathcal{A}$ be the collection of axis-aligned rectangles. Given sample $S = S_1 \cup S_2$, consider the following finite subset of $\mathcal{A}$ defined by

$$\mathcal{H}_{\text{DR}}(S) \triangleq \{R(y_i, y_j, x_m, x_n) : (x_m, y_i), (x_n, y_j) \in S \text{ or } (x_m, y_j), (x_n, y_i) \in S\} \quad (29)$$

where $R(y_i, y_j, x_m, x_n)$ is the axis-aligned rectangle defined as in (14). See Fig. 7.

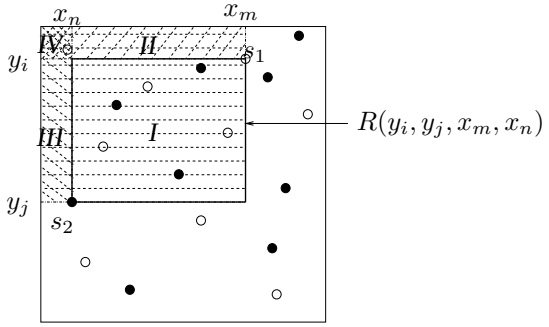


Fig. 7. Members of $\mathcal{H}_{\text{DR}}$

Proposition 4.8:

$$\text{VC-d}(\mathcal{H}_{\text{DR}}) = 2.$$

Proof:

It is easy to see that VC-d($\mathcal{H}_{\text{DR}}$) ≥ 2 because any set of two points can be shattered (a singleton also belongs to $\mathcal{H}_{\text{DR}}$).

For any set S of 3 points, *i.e.*, $S = \{s_1, s_2, s_3\}$. If there is no set in $\mathcal{H}_{\text{DR}}$ containing S, then S is not shatterable. Otherwise, let s_1, s_2 be the points defining such a set, *i.e.*, the axis-aligned rectangle with diagonal vertices s_1, s_2 contains S. Then $\{s_1, s_2\}$ cannot be shattered because the only way to shatter it is by the axis-aligned rectangle with s_1, s_2 as diagonal vertices, but this rectangle also contains s_3. Hence VC-d($\mathcal{H}_{\text{DR}}$) ≤ 2.

■

$\mathcal{H}_{\text{DR}}$ is not complete w.r.t. $\mathcal{A}$. However, by the same argument as in Algorithm SCD, we see that if the probability distributions are such that any disk with positive measure has positive probability, the loss of performance vanishes as sample size goes to infinity.

Algorithm SDR($d_{\mathcal{A}}$) and Algorithm SDR($\phi_{\mathcal{A}}$) share the following steps:

Initially, the algorithm builds two matrices C_1 and C_2 to store the empirical cdf(cumulative distribution function) of S_1 and S_2. Specifically, assuming $x_1 \leq x_2 \leq \ldots \leq x_M$, and $y_1 \leq y_2 \leq \ldots \leq y_M$, define

$$C_k(j, i) \triangleq |S_k \cap R(0, y_j, 0, x_i)|/|S_k|, \; k = 1, 2.$$

Construct C_1 and C_2 recursively:

(i) Sort S by the abscissa and ordinates respectively;

Define function $\delta_k : \{1, \ldots, M\} \to \{0, 1\}$, $k = 1, 2$,

$\delta_k(j) = 1$ if the sensor with ordinate y_j belongs to S_k.

Define function $g : \{1, \ldots, M\} \to \{1, \ldots, M\}$,

$$g(j) = i \text{ if } (x_i, y_j) \in S.$$

(ii) Compute the first row:

$$C_k(1, m) = \frac{\delta_k(1)}{|S_k|} \text{ if } m \geq g(1) \quad (30)$$

$$= 0 \text{ otherwise} \quad (31)$$

$k \in \{1, 2\}$, $m = 1, \ldots, M$.

(iii) Compute the j-th row, $j = 2, \ldots, M$:

$$C_k(j, m) = C_k(j-1, m) + \frac{\delta_k(j)}{|S_k|} \text{ if } m \geq g(j) \quad (32)$$

$$= C_k(j-1, m) \text{ otherwise} \quad (33)$$

$k \in \{1, 2\}$, $m = 1, \ldots, M$.

Then compute empirical probabilities for members of $\mathcal{H}_{\text{DR}}$: for every rectangle $R(y_i, y_j, x_m, x_n) \in \mathcal{H}_{\text{DR}}$, $i \leq j, m \leq n$, its empirical probabilities are given by

$$S_k(R(y_i, y_j, x_m, x_n)) = \begin{cases} S'_k(R(y_i, y_j, x_m, x_n)) + \frac{\delta_k(i)}{|S_k|} & \text{if } (x_m, y_i) \in S \\ S'_k(R(y_i, y_j, x_m, x_n)) + \frac{\delta_k(i)+\delta_k(j)}{|S_k|} & \text{o.w.} \end{cases} \quad (34)$$

where

$$S'_k(R(y_i, y_j, x_m, x_n)) = C_k(j, n) - C_k(i, n) - C_k(j, m) + C_k(i, m), \quad (35)$$

$k \in \{1, 2\}$. As seen in Fig. 7, the probability of the bold rectangle is the probability of *I* minus that of *II*, minus *III*, and plus *IV*, and we need the amendments to take care of boundary points.

Then Algorithm SDR($d_{\mathcal{A}}$) computes

$$\max_{R \in \mathcal{H}_{\text{DR}}} |S_1(R) - S_2(R)|,$$

and Algorithm SDR($\phi_{\mathcal{A}}$) computes

$$\max_{R \in \mathcal{H}_{\text{DR}}} \frac{|S_1(R) - S_2(R)|}{\sqrt{\frac{S_1(R)+S_2(R)}{2}}}.$$

Algorithm SDR($d_{\mathcal{A}}$) and Algorithm SDR($\phi_{\mathcal{A}}$) both have complexity $O(M^2)$ because constructing matrices C_1 and C_2 takes $O(M^2)$ steps and the search exhausts the $O(M^2)$ rectangles in $\mathcal{H}_{\text{DR}}$. Note that this algorithm requires a substantial amount of space: $O(M^2)$, which is due to the space to store C_1 and C_2.

V. Simulation

A. Simulation Setup

In the simulation, we consider the case when the distribution of collected sensors is a mixture of 2D uniform distributions, one on an $s \times s$ square $\mathcal{D}$ and the other centered at $\mathbf{x}_0 \in \mathcal{D}$ with radius r. Specifically, the PDF of the 2D random vector $\mathbf{x}$ is given by

$$p_{\mathbf{x}_0}(\mathbf{x}) = \begin{cases} \frac{p}{\pi r^2 p + (s^2 - \pi r^2) q} & \mathbf{x} \in \mathcal{D}, ||\mathbf{x} - \mathbf{x}_0|| \le r \\ \frac{q}{\pi r^2 p + (s^2 - \pi r^2) q} & \mathbf{x} \in \mathcal{D}, ||\mathbf{x} - \mathbf{x}_0|| > r \\ 0 & \text{otherwise} \end{cases}$$

where $\mathbf{x}_0$, p, q, and r are parameters, $0 < r << s$ and $0 \le q < p \le 1$.

This model corresponds to the scenario when sensors are uniformly distributed in $\mathcal{D}$, and a sensor is alarmed with probability p if it is within distance r from $\mathbf{x}_0 \in \mathcal{D}$ and q if it falls outside this distance. If we view the disk $\{\mathbf{x} \in \mathcal{D} : ||\mathbf{x} - \mathbf{x}_0|| \le r\}$ as the area where a noiseless sensor measurement should be "alarm" and the area outside this disk be where a noiseless measurement should be "non-alarm", then $1 - p$ is the (uniform) miss detection probability and q is the (uniform) false alarm probability at sensors.

Under hypothesis $\mathcal{H}_0$, two sets of sample points are drawn i.i.d. from the same $p_{\mathbf{x}_0}$; under $\mathcal{H}_1$, one set of sample points are drawn from $p_{\mathbf{x}_0}$, and the other set of sample points are drawn independently from $p_{\mathbf{x}_0'}$ for some other center $\mathbf{x}_0'$.

B. Detector Sensitivity

We consider Neyman-Pearson detection with detector size α, and choose detection threshold according to (12) to guarantee that the detector's false alarm will not exceed α.

Recalling that $\epsilon(n)$ measures detector sensitivity, we examine the relation between $\epsilon(n)$, the VC-dimension and the distance measure. Note that for fixed false alarm, we need more sample points to achieve the same threshold for a test searching in a class of larger VC-dimension. For searches in classes of the same VC-dimension, the test using relative $\mathcal{A}$-distance needs less sample points to achieve the same threshold than the one using $\mathcal{A}$-distance. See Fig.8.

Fig.9 shows that the detection threshold is not sensitive to the maximum false alarm α. We see that given a certain sample size, a detector with a larger size would not have a much smaller detection threshold. Hence increasing the sample size is usually the only way to improve the accuracy of the detector.

C. Detector Performance

We focus on miss detection in our Monte Carlo simulations. Fig. 10 and Fig. 11 show the miss detection probability vs. sample size. We observe that there is a threshold sample size

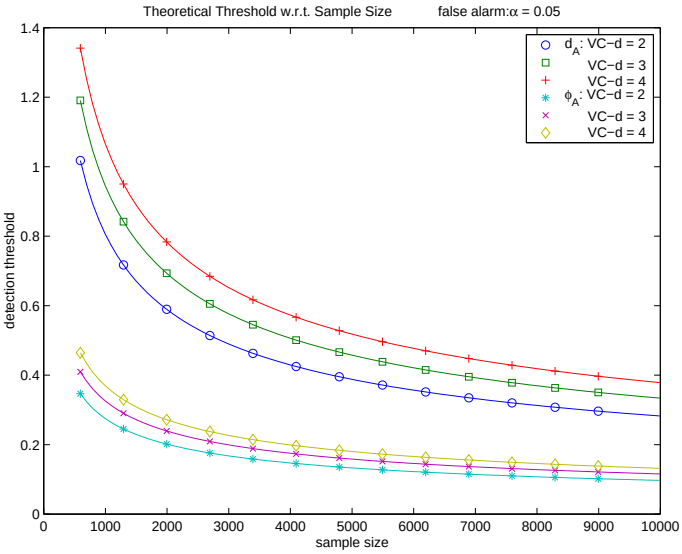


Fig. 8. Detection threshold as a function of the sample size for different VC-dimension's

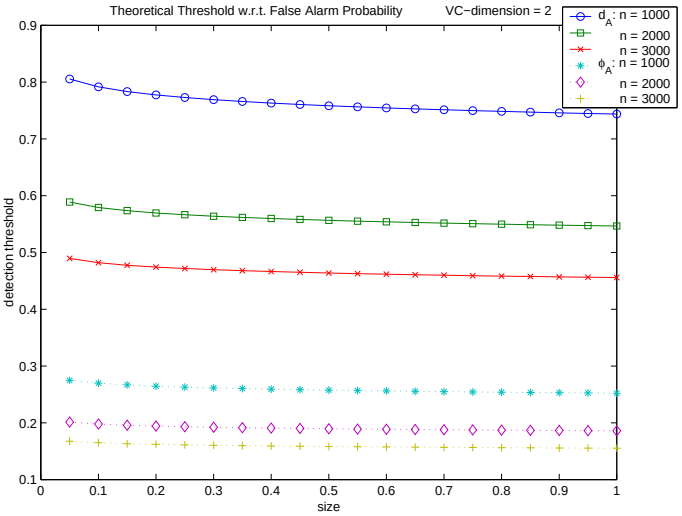


Fig. 9. Detection threshold as a function of the detector size for different sample sizes

beyond which the miss detection probability drops sharply. This can be explained using Theorem 3.1, which states that the upper bound on miss detection probability begins to drop when $\epsilon(n) < d_{\mathcal{A}}(P_1, P_2)$ for $\delta_{d_{\mathcal{A}}}$ or $\epsilon(n) < \phi_{\mathcal{A}}(P_1, P_2)$ for $\delta_{\phi_{\mathcal{A}}}$, and once it starts to drop, it drops exponentially. A heuristic argument on the minimum sample size would be that the sample size n should be s.t.

$$\epsilon(n) = \sqrt{\frac{32}{n} \log \frac{8(2n+1)^d}{\alpha}} \le d_{\mathcal{A}}(P_1, P_2) \text{ for } \delta_{d_{\mathcal{A}}} \tag{36}$$

$$\epsilon(n) = \sqrt{\frac{4}{n} \log \frac{2(2n+1)^d}{\alpha}} \le \phi_{\mathcal{A}}(P_1, P_2) \text{ for } \delta_{\phi_{\mathcal{A}}} \tag{37}$$

If we know P_1 and P_2, we can calculate $d_{\mathcal{A}}(P_1, P_2)$ and $\phi_{\mathcal{A}}(P_{1,2})$ to obtain a lower bound on n by solving

the inequalities (36) and (37). An observation is that this estimation is close to the minimum sample size required in the simulation. For example, in our simulation setup, the estimated minimum sample sizes for Algorithm SAS and SCD using $\mathcal{A}$-distance metric are both 2725, and that for SCD using relative $\mathcal{A}$-distance metric is 53. As indicated in Fig. 10 and Fig. 11, they all agree well to the sharp drop in missing detection probabilities.

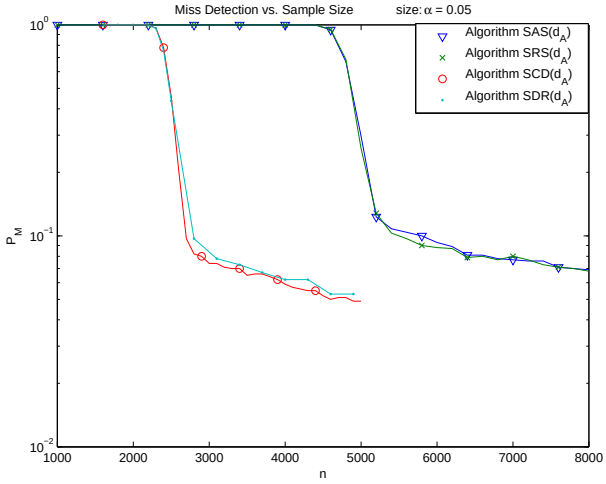


Fig. 10. Miss detection probability of $\delta_{d_{\mathcal{A}}}$ as a function of the sample size: simulation results. Here $p = 0.98$, $q = 0.02$, $r = s/12$. Use 1000 Monte Carlo runs.

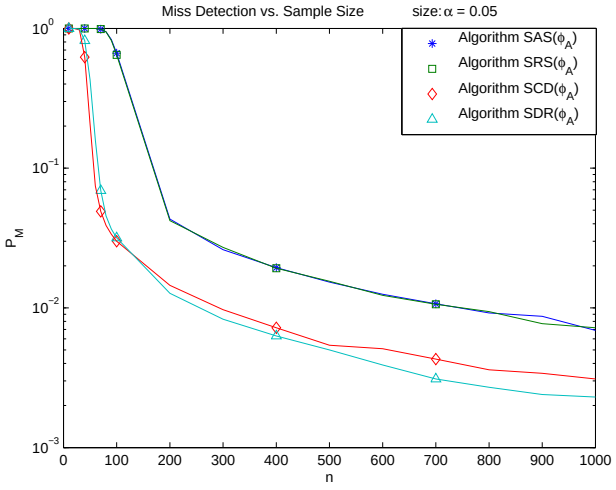


Fig. 11. Miss detection probability of $\delta_{\phi_{\mathcal{A}}}$ as a function of the sample size: simulation results. Here $p = 0.98$, $q = 0.02$, $r = s/12$. Use 10000 Monte Carlo runs.

As expected, both threshold and miss detection probability are decreasing functions of sample size, which reflects a trade-off between detection precision and sampling time, energy consumed and data processing expense.

We also plot the detection probability w.r.t. the size of the detector. See Fig. 12 and Fig. 13. The plot shows the detection probability does not increase significantly with the increase of the detector size, which is expected because the size affects detection probability only through the threshold, and the threshold is not sensitive to the change of size (see Fig. 9).

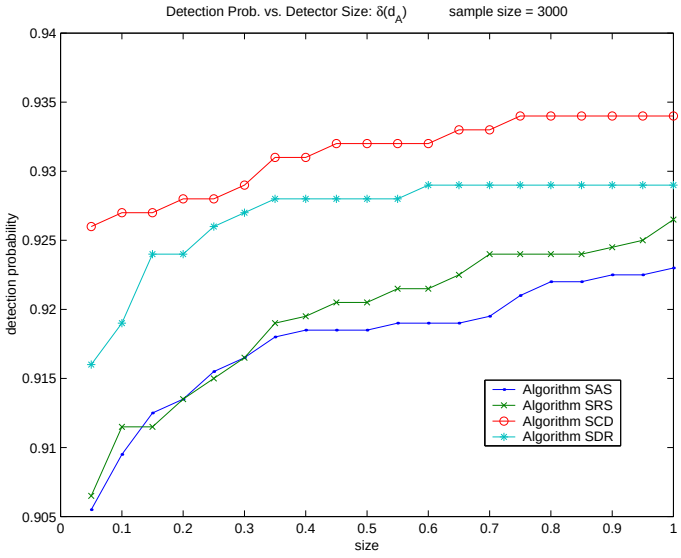


Fig. 12. Detection probability of $\delta_{d_{\mathcal{A}}}$ as a function of detector size, 1000 Monte Carlo runs.

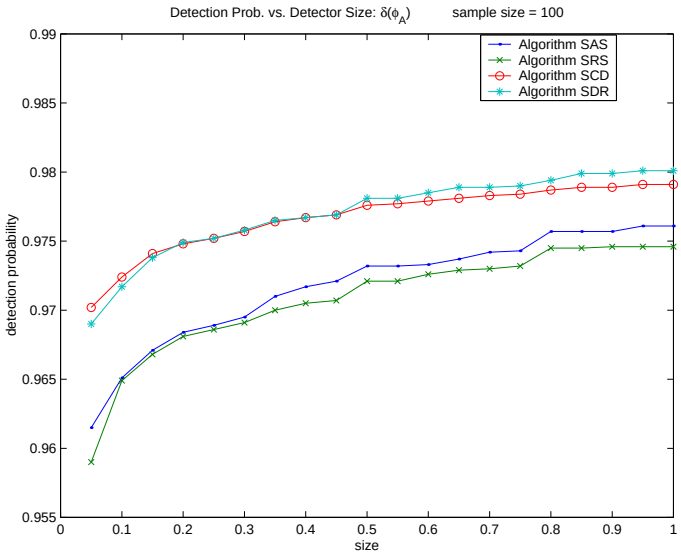


Fig. 13. Detection probability of $\delta_{\phi_{\mathcal{A}}}$ as a function of detector size, 10000 Monte Carlo runs.

Note that by choosing the threshold from the upper bound in (38) and (41), we only guarantee the false alarm is upper bounded by α. Our simulation shows the actual false alarm probability can be much less than the size of the detector[8], which implies that the theoretical threshold is a loose upper bound of the actual minimum threshold needed to guarantee the required detector size. This is because of the nonparametric

[8]For example, in our simulation of Algorithm SAS and SRS, for sample size up to $10,000$ using 1000 Monte Carlo runs, we encounter no false alarm at all.

nature of the theoretical threshold. This threshold is proved to satisfy the size constraint under arbitrary distributions by the Vapnik-Chervonenkis Theory. Therefore for a given distribution, this threshold may be loose.

For comparison among the algorithms, an obvious observation is that $\delta_{\phi_{\mathcal{A}}}$ outperforms $\delta_{d_{\mathcal{A}}}$ in detection probability. This is because on one hand, given n and α, using (36,37) to choose threshold yields that $\epsilon(n)$ for $\phi_{\mathcal{A}}$ is $1/2\sqrt{2}$ smaller than that for $d_{\mathcal{A}}$; on the other hand, we have $\phi_{\mathcal{A}}(S_1, S_2) \geq d_{\mathcal{A}}(S_1, S_2)$. Therefore in our simulation it is easier for algorithms using statistic $\phi_{\mathcal{A}}(S_1, S_2)$ to detect a change. However, this is caused by the specific way to decide the detection threshold, and does not imply that $\delta_{\phi_{\mathcal{A}}}$ is uniformly better than $\delta_{d_{\mathcal{A}}}$.

An intuitive guideline in algorithm design is that the better sets in $\mathcal{A}$ separate the probability mass in P_1 and P_2 and the simpler $\mathcal{A}$ is, the better the detector performance is, e.g. Algorithm SCD performs better than Algorithm SAS and SRS. Moreover, we can introduce random factors into the algorithm to make it more robust, e.g. we randomize SAS to be SRS so as to make it independent of the direction in which change occurs.

VI. Extension to Finite-level Sensor Measurements

We have presented our results based on collecting sensor locations of sensors with the same report (*i.e.,* "alarm"). Extension can be made to applications with finite-level sensor measurements.

Without loss of generality, let each sensor report either it is alarmed (say, measurement level 1) or it is not alarmed (level 0). In such a case, the ith data collection is modelled by probability space $(X \times \{0, 1\}, \mathcal{F}, P_i)$ where $\mathcal{F}$ is a σ-field on $X \times \{0, 1\}$. Let random variable $\mathbf{x} \in X$ denote the sensor location, and $L \in \{0, 1\}$ denote the sensor report. In the ith collection, $(\mathbf{x}, L)$ has joint distribution P_i, and the location of alarmed sensors has conditional distribution $P_i|_{L=1}$. It is easy to see that there are cases when P_i changes but $P_i|_{L=1}$ does not. Hence by collecting both types of sensor reports, we are able to detect a wider range of changes.

To apply the algorithms presented previously, choose class $\mathcal{A}'$ to be the collection of sets from $\mathcal{A}$ in either 0-plane or 1-plane, *i.e.,* $\mathcal{A}' = \mathcal{A} \times \{0, 1\}$. For instance, the collection of planar disks becomes the collection of planar disks with either measurement 0 or measurement 1. Algorithms should be applied to both 0-plane and 1-plane and we choose the larger as the test statistics $d_{\mathcal{A}}(S_1, S_2)$ or $\phi_{\mathcal{A}}(S_1, S_2)$. The detection and estimation performance guarantee still holds, but note that the sample size now becomes the total number of sensor reports collected (rather than the number of alarms collected). Note that the VC-dimension of such a class $\mathcal{A}'$ remains the same as that of $\mathcal{A}$:

Proposition 6.1: For a class $\mathcal{A}$ of planar sets,

$$\text{VC-d}(\mathcal{A} \times \{0, 1\}) = \text{VC-d}(\mathcal{A}).$$

Proof:

It is easy to see that VC-d$(\mathcal{A} \times \{0, 1\}) \geq$VC-d$(\mathcal{A})$.

For any set S, if S contains points from different planes, S is not shatterable because no set in $\mathcal{A} \times \{0, 1\}$ contains points from different planes. If S only contains points in one plane, it is shatterable only if $|S| \leq$ VC-d$(\mathcal{A})$. Therefore, VC-d$(\mathcal{A} \times \{0, 1\}) \leq$VC-d$(\mathcal{A})$.

■

VII. Conclusion

We have presented in this paper a nonparametric approach to the detection of changes in the distribution of alarmed sensors. We have provided exponential bounds for the miss detection and false alarm probabilities. The error exponents of these probabilities provide useful guideline for determining the number of sample points required.

We have also proposed several nonparametric change detection and estimation algorithms. Here we have aimed at reducing the computation complexity while preserving the theoretical performance guarantee by using recursive search strategies that reuse earlier computations, which gives us two near linear-complexity algorithms SAS and SRS. The more expensive algorithms SCD and SDR also have their roles, despite their near square cost, especially in detecting changes of highly clustered distributions. This is because the search classes in Algorithm SCD and SDR may yield larger distance than the more simplified classes, which in turn gives larger error exponents as indicated in Theorem 3.1. Moreover, Algorithm SCD is much more efficient than the exhaustive algorithm SPD with complexity $O(M^4)$, and Algorithm SDR also improves the complexity of its exhaustive counterpart Algorithm SAR significantly. Complexities of different algorithms presented so far are summed up in the following table.

TABLE I
Time Complexity Comparison

	$d_{\mathcal{A}}$	$\phi_{\mathcal{A}}$
SPD	$O(M^4)$	$O(M^4)$
SCD	$O(M^2 \log M)$	$O(M^2 \log M)$
SAR	$O(M^3)$	$O(M^4)$
SDR	$O(M^2)$	$O(M^2)$
SAS	$O(M \log M)$	$O(M^2)$
SRS	$O(M \log M)$	$O(M^2)$

Besides running time, one may also care about the amount of storage used for executing the algorithms. Obviously $O(M)$ space is needed to store S_1 and S_2, and the extra space needed scales as follows:

TABLE II
Space Complexity Comparison

	$d_{\mathcal{A}}$	$\phi_{\mathcal{A}}$
SPD	$O(1)$	$O(1)$
SCD	$O(1)$	$O(1)$
SAR	$O(1)$	$O(M)$
SDR	$O(M^2)$	$O(M^2)$
SAS	$O(1)$	$O(M)$
SRS	$O(1)$	$O(M)$

Comparing these tables, one can see the time-space tradeoff in algorithm design. For example, although Algorithm SDR

has comparable running time with Algorithm SCD, it requires much more space to execute, i.e. $O(M^2)$ instead of $O(1)$. The choice of algorithm should be a trade-off between running time, space requirement and detection performance, with the significance of each highly dependent on applications.

One should be further cautioned that the techniques considered in this paper typically require a large number of sample points. Since no information about the distribution is used, and the performance guarantee must hold for all distributions, bounds derived here are conservative. While in this paper we adhere to the principle of nonparametric approach, the incorporation of certain prior knowledge about the distribution, in the selection of $\mathcal{A}$ for example, would lead to more effective detection and estimation schemes in practice.

APPENDIX

A. Proof of Theorem 3.1

Proof: We first prove the theorem for detectors using the $\mathcal{A}$-distance metric $d_{\mathcal{A}}(S_1, S_2) = \sup_{A\in\mathcal{A}} |S_1(A) - S_2(A)|$. From [12], we have

$$\Pr\{\exists A \in \mathcal{A}, ||P_1(A) - P_2(A)| - |S_1(A) - S_2(A)|| > \epsilon\} \le 8(2n+1)^d e^{-n\epsilon^2/32} \quad (38)$$

Under H_0, $P_1 = P_2$, and the false alarm probability satisfies

$$\begin{aligned} P_F(\delta) &= \Pr\{d_{\mathcal{A}}(S_1, S_2) > \epsilon; \mathcal{H}_0\} \\ &= \Pr\{\exists A \in \mathcal{A}, |S_1(A) - S_2(A)| > \epsilon; \mathcal{H}_0\} \\ &= \Pr\{\exists A \in \mathcal{A}, ||P_1(A) - P_2(A)| \\ &\quad -|S_1(A) - S_2(A)|| > \epsilon; \mathcal{H}_0\} \\ &\le 8(2n+1)^d e^{-n\epsilon^2/32} \end{aligned} \quad (39)$$

where inequality (39) follows from (38).

For the miss probability, let $A^* = \arg\max_{A\in\mathcal{A}} |P_1(A) - P_2(A)|$.

$$\begin{aligned} P_M(\delta, P_1, P_2) &= \Pr\{d_{\mathcal{A}}(S_1, S_2) \le \epsilon; P_1, P_2\} \\ &\le \Pr\{|S_1(A^*) - S_2(A^*)| \le \epsilon; P_1, P_2\} \\ &\le \Pr\{||P_1(A^*) - P_2(A^*)| \\ &\quad -|S_1(A^*) - S_2(A^*)|| \\ &\quad \ge ||P_1(A^*) - P_2(A^*)| - \epsilon|\,; P_1, P_2\} \\ &\le 8(2n+1)^d e^{-n[|P_1(A^*)-P_2(A^*)|-\epsilon]^2/32} \end{aligned} \quad (40)$$

Now consider relative distance. The proof for relative distance metric goes line by line as that for the non-relative metric, replacing inequality (38) with the following results from [12],

$$P^{2n}(\phi_{\mathcal{A}}(S_1, S_2) > \epsilon) \le 2(2n+1)^d e^{-n\epsilon^2/4} \quad (41)$$

$$P^{2n}[|\phi_{\mathcal{A}}(P_1, P_2) - \phi_{\mathcal{A}}(S_1, S_2)| > \epsilon] \le 16(2n+1)^d e^{-n\epsilon^2/16} \quad (42)$$

We have

$$P_F(\delta) \le 2(2n+1)^d e^{-n\epsilon^2/4} \quad (43)$$

$$P_M(\delta, P_1, P_2) \le 16(2n+1)^d e^{-n[\phi_{\mathcal{A}}(P_1,P_2)-\epsilon]^2/16} \quad (44)$$

■

B. Proof of Theorem 3.2

Proof: Let $VC\text{-}d(\mathcal{A}) = d < \infty$. We first prove the theorem for $\mathcal{A}$-distance.

Let

$$A = \arg\max_{B\in\mathcal{A}} |P_1(B) - P_2(B)|,$$

and define η to be

$$\eta \triangleq |P_1(A) - P_2(A)| - \sup_{\substack{B\in\mathcal{A} \\ B\neq A}} |P_1(B) - P_2(B)|.$$

The uniqueness of A says $\eta > 0$.

By results of [12], we have

$$\Pr\{\sup_{B\in\mathcal{A}} ||P_1(B) - P_2(B)| - |S_1(B) - S_2(B)|| \le \frac{\eta}{3}\} \ge 1 - 8(2n+1)^d e^{-n\eta^2/288}.$$

So with probability $\ge 1 - 8(2n+1)^d e^{-n\eta^2/288}$,

$$\begin{aligned} &|S_1(A) - S_2(A)| - \sup_{\substack{B\in\mathcal{A} \\ B\neq A}} |S_1(B) - S_2(B)| \\ \ge\ & |P_1(A) - P_2(A)| - \sup_{\substack{B\in\mathcal{A} \\ B\neq A}} |P_1(B) - P_2(B)| \\ & -||S_1(A) - S_2(A)| - |P_1(A) - P_2(A)|| \\ & -|\sup_{\substack{B\in\mathcal{A} \\ B\neq A}} |S_1(B) - S_2(B)| - \sup_{\substack{B\in\mathcal{A} \\ B\neq A}} |P_1(B) - P_2(B)|| \quad (45) \\ \ge\ & \eta - 2\sup_{B\in\mathcal{A}} ||P_1(B) - P_2(B)| - |S_1(B) - S_2(B)|| \quad (46) \\ \ge\ & \tfrac{\eta}{3} \quad (47) \end{aligned}$$

That is,

$$\Pr\{A = \arg\max_{B\in\mathcal{A}} |S_1(B) - S_2(B)|\} \ge 1 - 8(2n+1)^d e^{-n\eta^2/288}.$$

Now let $n \to \infty$,

$$\lim_{n\to\infty} \Pr\{A = \arg\max_{B\in\mathcal{A}} |S_1(B) - S_2(B)|\} = 1.$$

For relative $\mathcal{A}$-distance, let

$$A = \arg\max_{B\in\mathcal{A}} \frac{|P_1(B) - P_2(B)|}{\sqrt{\frac{P_1(B)+P_2(B)}{2}}}.$$

Let

$$\eta \triangleq f_\phi(P_1(A), P_2(A)) - \sup_{\substack{B\in\mathcal{A} \\ B\neq A}} f_\phi(P_1(B), P_2(B)).$$

The uniqueness of A says $\eta > 0$.

In [9] it is proved that $f_\phi(x, y)$ is a metric on $[0, 1]$.

The proof is similar to that of $\mathcal{A}$-distance. By [12] we have

$$\Pr(\sup_{B\in\mathcal{A}} f_\phi(S_i(B), P_i(B)) \le \frac{\eta}{5}) \ge 1 - 8(2n+1)^d e^{-n\eta^2/100}, \quad i = 1, 2.$$

So with probability $\ge [1 - 8(2n+1)^d e^{-n\eta^2/100}]^2$, we have

$$f_\phi(S_1(A), S_2(A)) - \sup_{\substack{B\in\mathcal{A} \\ B\neq A}} f_\phi(S_1(B), S_2(B))$$

$$\geq f_\phi(P_1(A),P_2(A)) - \sup_{\substack{B\in\mathcal{A}\\B\neq A}} f_\phi(P_1(B),P_2(B)) - |f_\phi(P_1(A),P_2(A)) - f_\phi(S_1(A),S_2(A))| - |\sup_{\substack{B\in\mathcal{A}\\B\neq A}} f_\phi(P_1(B),P_2(B)) - \sup_{\substack{B\in\mathcal{A}\\B\neq A}} f_\phi(S_1(B),S_2(B))| \quad (48)$$

$$\geq \eta - f_\phi(P_1(A),S_1(A)) - f_\phi(P_2(A),S_2(A)) - \sup_{\substack{B\in\mathcal{A}\\B\neq A}} f_\phi(P_1(B),S_1(B)) - \sup_{\substack{B\in\mathcal{A}\\B\neq A}} f_\phi(P_2(B),S_2(B)) \quad (49)$$

$$\geq \eta - 2\sup_{B\in\mathcal{A}} f_\phi(P_1(B),S_1(B)) - 2\sup_{B\in\mathcal{A}} f_\phi(P_2(B),S_2(B)) \quad (50)$$

$$\geq \frac{\eta}{5} \quad (51)$$

That is,

$$\Pr\{A = \arg\max_{B\in\mathcal{A}} \frac{|S_1(B)-S_2(B)|}{\sqrt{\frac{S_1(B)+S_2(B)}{2}}}\} \geq [1-8(2n+1)^d e^{-n\eta^2/100}]^2.$$

Now let $n\to\infty$, the proof completes.

■

REFERENCES

[1] L. Tong, Q. Zhao, and S. Adireddy, "Sensor Networks with Mobile Agents," in *Proc. 2003 Intl. Symp. Military Communications*, (Boston, MA), Oct. 2003.

[2] V. Vapnik and A. Y. Chervonenkis, "On the uniform convergence of relative frequencie of events to their probabilities," *Theory of Probability and its Applications*, vol. 16, pp. 264–280, 1971.

[3] S. Ben-David, T. He, and L. Tong, "Non-Parametric Approach to Change Detection and Estimation in Large Scale Sensor Networks," in *Proceedings of the 2004 Conference on Information Sciences and Systems*, (Princeton, NJ), March 2004.

[4] N. Patwari, A. O. Hero, and B. M. Sadler, "Hierarchical censoring sensors for change detection," in *2003 IEEE Workshop on Statistical Signal Processing*, (St. Louis, MO), pp. 21–24, September 2003.

[5] Y. Hong and A. Scaglione, "Distributed change detection in large scale sensor networks through the synchronization of pulse-coupled oscillators," in *Proc. Intl. Conf. Acoust., Speech, and Signal Processing*, (Montreal, Canada), pp. 869 – 872, May 2004.

[6] J. D. Gibbons and S. Chakraborti, *Nonparametric Statistical Inference*. Marcel Dekker, 2003.

[7] M. Hollander and D. A. Wolfe, *Nonparametric Statistical Methods*. Wiley Interscience, 1973.

[8] D. J. Sheskin, *Handbook of Parametric and Nonparametric Statistical Procedures*. Chapman & Hall/CRC, 2004. 3rd Ed.

[9] S. Ben-David, J. Gehrke, and D. Kifer, "Detecting Change in Data Streams," in *Proc. 2004 VLDB Conference*, (Toronto, Canada), 2004.

[10] P. Gupta and P. R. Kumar, "The capacity of wireless networks," *IEEE Trans. Inform. Theory*, vol. 46, pp. 388–404, March 2000.

[11] T. He and L. Tong, "An Almost Surely Complete Subset of Planar Disks," Tech. Rep. ACSP-TR-04-05-01, Cornell University, April 2005. `http://acsp.ece.cornell.edu/pubR.html`.

[12] T. He and L. Tong, "On $\mathcal{A}$-distance and Relative $\mathcal{A}$-distance," Tech. Rep. ACSP-TR-08-04-02, Cornell University, August 2004. `http://acsp.ece.cornell.edu/pubR.html`.