Collaborative Signal Processing for Distributed Classification in Sensor Networks

Ashwin D'Costa and Akbar M. Sayeed*

Electrical and Computer Engineering
University of Wisconsin-Madison
dcosta@cae.wisc.edu, akbar@engr.wisc.edu

Abstract. Sensor networks provide virtual snapshots of the physical world via distributed wireless nodes that can sense in different modalities, such as acoustic and seismic. Classification of objects moving through the sensor field is an important application that requires collaborative signal processing (CSP) between nodes. Given the limited resources of nodes, a key constraint is to exchange the least amount of information between them to achieve desired performance. Two main forms of CSP are possible. Data fusion – exchange of low dimensional feature vectors – is needed between correlated nodes, in general, for optimal performance. Decision fusion – exchange of likelihood values – is sufficient between independent nodes. Decision fusion is generally preferable due to its lower communication and computational burden. We study CSP of multiple node measurements for classification, each measurement modeled as a Gaussian (target) signal vector corrupted by additive white Gaussian noise. The measurements are partitioned into groups. The signal components within each group are perfectly correlated whereas they vary independently between groups. Three classifiers are compared: the optimal maximum-likelihood classifier, a data-averaging classifier that treats all measurements as correlated, and a decision-fusion classifier that treats them all as independent. Analytical and numerical results based on real data are provided to compare the performance of the three CSP classifiers. Entropy comparison between data- and decision-fusion is also provided to quantify the lower communication burden in decision fusion. Our results indicate that the sub-optimal decision fusion classifier, that is most attractive in the context of sensor networks, is also a robust choice from a decision-theoretic viewpoint.

1 Introduction

Wireless sensor networks are an emerging technology for monitoring the physical world with a densely distributed network of wireless nodes [1]. Each node has limited communication and computation ability and can sense the environment in a variety of modalities, such as acoustic, seismic, and infra red [1, 2, 3]. A wide variety of applications are being envisioned for sensor networks, including disaster

* This work was supported by DARPA SensIT program under Grant F30602-00-2-0555.

relief, border monitoring, condition-based machine monitoring, and surveillance in battlefield scenarios. Detection and classification of objects moving through the sensor field is an important task in many envisioned applications. Exchange of sensor information between different nodes in the vicinity of the object is necessary for reliable execution of such tasks due to a variety of reasons, including limited (local) information gathered by each node, variability in operating conditions, and node failure. Consequently, development of theory and methods for collaborative signal processing (CSP) of the data collected by different nodes is a key research area for realizing the vision of sensor networks.

The CSP algorithms have to be developed under the constraints imposed by the limited communication and computational abilities of the nodes as well as their finite battery life. A key goal of CSP algorithms in sensor networks is to exchange the least amount of data between nodes to attain a desired level of performance. In this paper, with the above goal in mind, we investigate CSP algorithms for single-target classification based on multiple acoustic measurements at different nodes. The numerical results presented here are based on real data collected in the DARPA SensIT program.

Some form of region-based processing is attractive in sensor networks in order to facilitate CSP between nodes and also for efficient routing of information in applications involving tracking of moving targets [3]. Typically, the nodes in the network are partitioned into a number of regions and a manager node is designated within each region to facilitate CSP between the nodes in the region and for communication of information from one region to another. Single target classification and tracking generally involves the following steps [3]:

1. **Target detection and data collection.** A target is detected in a particular region which becomes the active region. The detection of a target itself may involve CSP between nodes. For example, outputs of energy detectors may be communicated to the manager node to make the final decision. The nodes within the region that detect the target also collect time series data in different modalities that is communicated to the manager node for classification purposes.
2. **Target localization.** Target detection information (for example, the time of closest point of approach and energy detector outputs) from different nodes is used by the manager node to estimate the location of the target.
3. **Target location prediction.** Location estimates over a period of time are used by the manager node to predict target location at future time instants.
4. **Creation of new potential active regions.** When the target gets close to exiting the current region, the estimates of predicted target location are used to put new regions on alert for target detection.
5. **Determination of new active region.** Once the target is detected in a new region it becomes the new active region. The above four steps are repeated for target tracking through the sensor field.

In this paper, we are primarily concerned with CSP techniques for combining the data collected by different nodes for single-target classification within a par-

ticular active region. However, the basic principles apply to distributed decision making in sensor networks in general.

There are two main forms of information exchange between nodes dictated by the statistics of measured signals. If two nodes yield correlated measurements, *data fusion* is needed, in general, for optimal performance – exchange of (low-dimensional) feature vectors that yield sufficient information for desired classification performance. On the other hand, if two nodes yield independent measurements, *decision fusion* is sufficient – exchange of likelihood values (scalars) computed from individual measurements. In general, the measurements would exhibit a mixture of correlated and independent components and would require a combination of data and decision fusion between nodes. In the context of sensor networks, decision fusion is clearly the more attractive choice. First, it imposes a significantly lower communication burden on the network, compared to data fusion, since only scalars are transmitted to the manager node [3]. Second, it also imposes a lower computational burden compared to data fusion since lower dimensional data has to be jointly processed at the manager node.

In this paper, we investigate the design of CSP classifiers and assess their performance in an idealized abstraction of measurements from multiple nodes. We consider $K = Gn_G$ measurements corresponding to a particular event. The K measurements are split into G groups with n_G measurements in each group. The signal component in the n_G measurements in a particular group is identical (perfectly correlated), but it varies independently from group to group. We compare the performance of three classifiers: 1) the optimal maximum-likelihood (ML) classifier, 2) a sub-optimal (decision-fusion) classifier that treats all the measurements as independent, and 3) a sub-optimal (data-averaging) classifier that treats all the measurements as perfectly correlated. Our results indicate that the decision-fusion classifier is remarkably robust to the true statistical correlation between measurements. Thus, the decision-fusion classifier, that is the most attractive choice in view of the computational and communication constraints, is also a robust choice from a decision-theoretic viewpoint.

2 CSP Classifiers for Multiple Measurements

We consider Gaussian classifiers which assume that the underlying data has complex circular Gaussian statistics. The notation $\boldsymbol{x} \sim \mathcal{CN}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ means that $\mathrm{E}[\boldsymbol{x}] = \boldsymbol{\mu}$ and $\mathrm{E}[\boldsymbol{x}\boldsymbol{x}^H] = \boldsymbol{\Sigma}$ and $\mathrm{E}[\boldsymbol{x}\boldsymbol{x}^T] = \mathbf{0}$ (circular assumption). We first discuss the classifier structure for a single measurement and then generalize it to multiple measurements.

2.1 Single Measurement Classifier

Consider M target classes. Let $\boldsymbol{x}$ denote a complex-valued N-dimensional feature vector corresponding to a detected event. Under hypothesis $j = 1, \cdots, M$ (corresponding to j-th target class), $\boldsymbol{x}$ is modeled as

$$H_j \ : \ \boldsymbol{x} = \boldsymbol{s} + \boldsymbol{n} \ , \ j = 1, \cdots, M, \tag{1}$$

where $\boldsymbol{s} \sim \mathcal{CN}(\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)$ denotes the Gaussian signal component corresponding to the j-th class, and $\boldsymbol{n} \sim \mathcal{CN}(\boldsymbol{0}, \boldsymbol{I})$ denotes additive white Gaussian noise. A classifier C maps the event feature vector $\boldsymbol{x}$ to one of the target classes. We assume that all classes are equally likely. Thus, the optimal classifier is the maximum-likelihood (ML) classifier which takes the form [4]

$$C(\boldsymbol{x}) = \arg \max_{j \in \{1, \cdots, M\}} p_j(\boldsymbol{x}) \tag{2}$$

where $p_j(\boldsymbol{x})$ denotes the likelihood function for j-th class which takes the following form under the complex Gaussian assumption

$$p_j(\boldsymbol{x}) = \frac{1}{\pi^N |\boldsymbol{\Sigma}_j + \boldsymbol{I}|} e^{-(\boldsymbol{x}-\boldsymbol{\mu}_j)^H (\boldsymbol{\Sigma}_j + \boldsymbol{I})^{-1} (\boldsymbol{x}-\boldsymbol{\mu}_j)}. \tag{3}$$

In this paper, we assume zero-mean signals so that $\boldsymbol{\mu}_j = \boldsymbol{0}$ for all j and, thus, all information about the targets is contained in the covariance matrices $\boldsymbol{\Sigma}_j$. In practice, $\boldsymbol{\Sigma}_j$ has to be estimated from available training data. We assume that $\mathrm{tr}(\boldsymbol{\Sigma}_j)$ (signal energy) is the same for all j.

2.2 Multiple Measurement Classifier

Suppose that we have K measurements (in a given modality), $\{\boldsymbol{x}_1, \cdots, \boldsymbol{x}_K\}$, from different nodes available to us. We are interested in combining these measurements to achieve improved classification performance. Consider the concatenated NK-dimensional feature vector

$$\boldsymbol{x}^{cT} = [\boldsymbol{x}_1^T, \boldsymbol{x}_2^T, \cdots, \boldsymbol{x}_K^T] \tag{4}$$

which has the same form as (1) under different hypotheses except for the larger number of dimensions. The noise is still white but the signal correlation matrix under H_j can be partitioned as

$$\boldsymbol{\Sigma}_j^c = \begin{bmatrix} \boldsymbol{\Sigma}_{j,11} & \boldsymbol{\Sigma}_{j,12} & \cdots & \boldsymbol{\Sigma}_{j,1K} \\ \boldsymbol{\Sigma}_{j,21} & \boldsymbol{\Sigma}_{j,22} & \cdots & \boldsymbol{\Sigma}_{j,2K} \\ \vdots & \ddots & \vdots & \vdots \\ \boldsymbol{\Sigma}_{j,K1} & \boldsymbol{\Sigma}_{j,K2} & \cdots & \boldsymbol{\Sigma}_{j,KK} \end{bmatrix} \tag{5}$$

where $\boldsymbol{\Sigma}_{j,kk'} = \mathrm{E}[\boldsymbol{x}_k \boldsymbol{x}_{k'}^H]$ denotes the cross-covariance between the k-th and k'-th measurements. The optimal classifier operates on $\boldsymbol{x}^c$ and takes the form (2) with $p_j(\boldsymbol{x}^c)$ given by (3) by replacing $\boldsymbol{x}$ with $\boldsymbol{x}^c$ and $\boldsymbol{\Sigma}_j$ with $\boldsymbol{\Sigma}_j^c$.

2.3 A Simple Measurement Model

We now present a model for measurements that is used throughout the paper. Let $K = Gn_G$. Suppose that the signal component of $\boldsymbol{x}^c$ can be partitioned into G groups of n_G measurements each as

$$\boldsymbol{s}^{cT} = [\boldsymbol{s}_1^T, \cdots, \boldsymbol{s}_1^T, \boldsymbol{s}_2^T, \cdots, \boldsymbol{s}_2^T, \cdots, \boldsymbol{s}_G^T, \cdots, \boldsymbol{s}_G^T] \tag{6}$$

where the signal component of the n_G measurements in each group is identical and it varies independently from group to group. That is, $\{\boldsymbol{s}_1, \cdots, \boldsymbol{s}_G\}$ are i.i.d. according to $\mathcal{CN}(\boldsymbol{0}, \boldsymbol{\Sigma}_j)$ under H_j. The noise measurements, on the other hand, are independent across all measurements. The above signal model can capture a range of correlation between measurements. For $K = G$ ($n_G = 1$), all the measurements have independent signal components (no correlation), whereas for $K = n_G$ ($G = 1$), all the measurements have identical signal components (maximum correlation). We consider three classifiers based on the above model.

Optimum Classifier There are two sources of classification error: background noise and the inherent statistical variability in the signals captured by $\boldsymbol{\Sigma}_j$'s. The optimal classifier performs signal averaging within each group to reduce the noise variance and statistical averaging over the groups to reduce the inherent signal variations. The optimum classifier operates on the NG dimensional vector

$$\boldsymbol{y} = \begin{bmatrix} \boldsymbol{y}_1 \\ \vdots \\ \boldsymbol{y}_G \end{bmatrix} = \begin{bmatrix} \boldsymbol{s}_1 \\ \vdots \\ \boldsymbol{s}_G \end{bmatrix} + \begin{bmatrix} \boldsymbol{w}_1 \\ \vdots \\ \boldsymbol{w}_G \end{bmatrix} = \boldsymbol{s} + \boldsymbol{w} \tag{7}$$

where $\boldsymbol{y}_i$ are obtained by averaging the measurements in each group

$$\boldsymbol{y}_i = \frac{1}{n_G} \sum_{j=1}^{n_G} \boldsymbol{x}_{(i-1)G+j} = \boldsymbol{s}_i + \boldsymbol{w}_i \ , i = 1, \cdots, G. \tag{8}$$

Note that $\boldsymbol{w}_i$ are i.i.d. $\mathcal{CN}(\boldsymbol{0}, \boldsymbol{I}/n_G)$ due to signal averaging and $\boldsymbol{s}_i$ are i.i.d. $\mathcal{CN}(\boldsymbol{0}, \boldsymbol{\Sigma}_j)$ under H_j. It can be shown that the optimal classifier takes the form

$$C_{opt}(\boldsymbol{y}_1, \cdots, \boldsymbol{y}_G) = \arg \min_{j=1,\cdots,M} l_{opt,j}(\boldsymbol{y}_1, \cdots, \boldsymbol{y}_G) \tag{9}$$

where the (negative) log-likelihood function $l_{opt,j}(\boldsymbol{y})$ is given by

$$\begin{aligned} l_{opt,j}(\boldsymbol{y}) &= \log |\boldsymbol{\Sigma}_j + \boldsymbol{I}/n_G| + \frac{1}{G} \sum_{i=1}^{G} \boldsymbol{y}_i^H (\boldsymbol{\Sigma}_j + \boldsymbol{I}/n_G)^{-1} \boldsymbol{y}_i \\ &= \log |\boldsymbol{\Sigma}_j + \boldsymbol{I}/n_G| + \mathrm{tr}((\boldsymbol{\Sigma}_j + \boldsymbol{I}/n_G)^{-1} \hat{\boldsymbol{\Sigma}}_y) \end{aligned} \tag{10}$$

and $\hat{\boldsymbol{\Sigma}}_y = \frac{1}{G} \sum_{i=1}^{G} \boldsymbol{y}_i \boldsymbol{y}_i^H$ is the estimated data correlation matrix of $\{\boldsymbol{y}_i\}$.

It is insightful to consider two limiting cases. First, suppose that $K = n_G$ ($G = 1$ – perfectly correlated measurements). In the limit of large K

$$\lim_{K \to \infty} l_{opt,j}(\boldsymbol{y}) = \log |\boldsymbol{\Sigma}_j| + \boldsymbol{y}_1^H \boldsymbol{\Sigma}_j^{-1} \boldsymbol{y}_1 \tag{11}$$

which shows that noise is completely eliminated and the only remaining source of error is the inherent statistical variation in the signal. Now, suppose that $K = G$ ($n_G = 1$ – independent measurements). In the limit of large K

$$\lim_{K \to \infty} l_{opt,j}(\boldsymbol{y}) = \log |\boldsymbol{\Sigma}_j + \boldsymbol{I}| + \mathrm{tr}((\boldsymbol{\Sigma}_j + \boldsymbol{I})^{-1} \boldsymbol{\Sigma}_y) \tag{12}$$

where $\boldsymbol{\Sigma}_y = \boldsymbol{\Sigma}_m + \boldsymbol{I}$ under H_m. In this case, all statistical variation in the signal is removed due to ensemble averaging. However, there is a bias in the estimated data correlation (relative to $\boldsymbol{\Sigma}_j$) due to noise. Both data averaging (correlated measurements) and ensemble averaging (uncorrelated measurements) contribute to improved classifier performance. However, as we will see, ensemble averaging is more critical in the case of stochastic signals.

Decision-Fusion Classifier The sub-optimal decision-fusion classifier treats all measurements as independent:

$$\begin{aligned} C_{df}(\boldsymbol{x}_1, \cdots, \boldsymbol{x}_K) &= \arg \min_{j=1,\cdots,M} l_{df,j}(\boldsymbol{x}_1, \cdots, \boldsymbol{x}_K) \\ l_{df,j}(\boldsymbol{x}) &= \log|\boldsymbol{\Sigma}_j + \boldsymbol{I}| + \frac{1}{K}\sum_{i=1}^{K} \boldsymbol{x}_i^H (\boldsymbol{\Sigma}_j + \boldsymbol{I})^{-1} \boldsymbol{x}_i \\ &= \log|\boldsymbol{\Sigma}_j + \boldsymbol{I}| + \mathrm{tr}((\boldsymbol{\Sigma}_j + \boldsymbol{I})^{-1} \hat{\boldsymbol{\Sigma}}_x) \end{aligned} \tag{13}$$

where $\hat{\boldsymbol{\Sigma}}_x = \frac{1}{K}\sum_{i=1}^{K} \boldsymbol{x}_i \boldsymbol{x}_i^H$ is the estimated data correlation matrix of $\{\boldsymbol{x}_i\}$. Note that C_{opt} and C_{df} are identical for $K = G$ in the measurement model. Note also from (13) that the M scalars $\{\boldsymbol{x}_i^H (\boldsymbol{\Sigma}_j + \boldsymbol{I})^{-1} \boldsymbol{x}_i\}$ for $j = 1, \cdots, M$ need to be transmitted from the K nodes to the manager node. Thus, C_{df} imposes a much smaller communication (and computational) burden on the network since $M \ll N$ in general. We consider only soft decision fusion in this paper. Several other forms, including hard decision fusion, are also possible [5].

Data-Averaging Classifier The data-averaging classifier treats all measurements as correlated. It operates on the average of all measurements

$$\boldsymbol{y}_{da} = \frac{1}{K}\sum_{i=1}^{K} \boldsymbol{x}_i = \frac{1}{G}\sum_{i=1}^{G} \boldsymbol{y}_i = \boldsymbol{s}_{da} + \boldsymbol{w}_{da} \tag{14}$$

where $\boldsymbol{s}_{da} \sim \mathcal{CN}(\boldsymbol{0}, \boldsymbol{\Sigma}_j/G)$ under H_j and $\boldsymbol{w}_{da} \sim \mathcal{CN}(\boldsymbol{0}, \boldsymbol{I}/K)$ in the measurement model. The data-averaging classifier takes the form

$$\begin{aligned} C_{da}(\boldsymbol{y}_{da}) &= \arg \min_{j=1,\cdots,M} l_{da,j}(\boldsymbol{y}_{da}) \\ l_{da,j}(\boldsymbol{y}_{da}) &= \log|\boldsymbol{\Sigma}_j + \boldsymbol{I}/K| + \boldsymbol{y}_{da}^H (\boldsymbol{\Sigma}_j + \boldsymbol{I}/K)^{-1} \boldsymbol{y}_{da}. \end{aligned} \tag{15}$$

Note that C_{opt} and C_{da} are identical for $K = n_G$. All K measurements $\{\boldsymbol{x}_i\}$ have to be communicated to the manager node for the computation of C_{opt} and C_{da}. However, the computational burden of C_{da} is lower than that of C_{opt}.

3 Performance Analysis of the Three Classifiers

We analyze the performance of the three classifiers for $M = 2$ classes. The analysis for $M > 2$ is more involved and is the beyond the scope of this paper. Simple

union bounds can be obtained for $M > 2$ via the $M = 2$ analysis presented here. We also analyze the asymptotic performance in the limit of large number of independent measurements and also provide an entropy comparison between data and decision fusion.

We assess the performance in terms of the average probability of (correct) detection (PD)

$$PD_j = P(l_j < l_m \ , \ \forall m \neq j | H_j) \ , \ PD = \frac{1}{M} \sum_{j=1}^{M} PD_j \tag{16}$$

and the average probability of false alarm (PFA)

$$PFA_j = \frac{1}{M-1} \sum_{k=1, k \neq j}^{M} P(l_j < l_m, \forall m \neq j | H_k) \ , \ PFA = \frac{1}{M} \sum_{j=1}^{M} PFA_j. \tag{17}$$

For $M = 2$ the above expressions simplify to

$$PD_1 = P(l_1 < l_2 | H_1) = 1 - PFA_2 \ , \ PD_2 = P(l_2 < l_1 | H_2) = 1 - PFA_1 \tag{18}$$

$$PD = 1 - PFA. \tag{19}$$

Our analysis is based on a signal model in which the covariance matrices of different targets are simultaneously diagonalizable. This model is motivated in the next section and it also simplifies the exposition to gain insight into the performance of the classifiers.

3.1 Simultaneously Diagonalizable Classes

We assume that all the covariance matrices share the same eigenfunctions

$$\boldsymbol{\Sigma}_j = \boldsymbol{U} \boldsymbol{\Lambda}_j \boldsymbol{U}^H \ , \ j = 1, \cdots, M \tag{20}$$

where $\boldsymbol{U}$ represents the matrix of common (orthonormal) eigenvectors for all the classes – the different classes are characterized by the diagonal matrix of eigenvalues $\boldsymbol{\Lambda}_j = \mathrm{diag}(\lambda_j[1], \cdots, \lambda_j[N])$. One scenario in which this assumption is approximately valid is when the source signals for different targets can be modeled as stationary processes over the duration of the detected event. In such a case, choosing $\boldsymbol{U}$ as a discrete Fourier transform (DFT) matrix would serve as an approximate set of eigenfunctions [6]. The eigenvalues will then correspond to samples of the associated power spectral densities (PSD's). The numerical results in Section 4 are based on this assumption and rely on experimental data collected in the SensIT program. Note that given the knowledge of $\boldsymbol{\Lambda}_j$ in the measurement model of Section 2.3, a realization for the signal in the i-th group, from the j-th class, can be generated as

$$\boldsymbol{s}_i = \boldsymbol{U} \boldsymbol{\Lambda}_j^{1/2} \boldsymbol{z}_i \ , \ \boldsymbol{z}_i \sim \mathcal{CN}(\boldsymbol{0}, \boldsymbol{I}) \ , \ i = 1, \cdots, G. \tag{21}$$

The same $\boldsymbol{z}_i$ realization is used in the i-th group and it changes independently from group to group. We assume the above signal model and analyze the classifiers in the eigen (Fourier) domain so that $\{\boldsymbol{\Sigma}_j\}$ are replaced with $\{\boldsymbol{\Lambda}_j\}$.

3.2 Optimal Classifier

The test statistic for the optimal classifier takes the form

$$l_{opt,j}(\boldsymbol{y}_1, \cdots, \boldsymbol{y}_G) = \log|\widetilde{\boldsymbol{\Lambda}}_j| + \frac{1}{G}\sum_{i=1}^{G} \boldsymbol{y}_i^H \widetilde{\boldsymbol{\Lambda}}_j^{-1} \boldsymbol{y}_i \quad , \quad \widetilde{\boldsymbol{\Lambda}}_j = \boldsymbol{\Lambda}_j + \boldsymbol{I}/n_G \tag{22}$$

where $\boldsymbol{y}_i$ are i.i.d. according to $\mathcal{CN}(\boldsymbol{0}, \widetilde{\boldsymbol{\Lambda}}_j)$. Thus, $\boldsymbol{y}_i$ can be representated as $\boldsymbol{y}_i = \widetilde{\boldsymbol{\Lambda}}_j^{1/2} \boldsymbol{z}_i$ where $\{\boldsymbol{z}_i\}$ are i.i.d. $\mathcal{CN}(\boldsymbol{0}, \boldsymbol{I})$. Consider the computation of PD_1 first. It can be readily shown that under H_1

$$l_{opt,1} = \log|\widetilde{\boldsymbol{\Lambda}}_1| + \frac{1}{G}\sum_{i=1}^{G} \|\boldsymbol{z}_i\|^2 \quad , \quad l_{opt,2} = \log|\widetilde{\boldsymbol{\Lambda}}_2| + \frac{1}{G}\sum_{i=1}^{G} \boldsymbol{z}_i^H \widetilde{\boldsymbol{\Lambda}}_1 \widetilde{\boldsymbol{\Lambda}}_2^{-1} \boldsymbol{z}_i. \tag{23}$$

Thus,

$$PD_1 = P\left(\frac{1}{G}\sum_{i=1}^{G} \boldsymbol{z}_i^H \left[\boldsymbol{I} - \widetilde{\boldsymbol{\Lambda}}_1 \widetilde{\boldsymbol{\Lambda}}_2^{-1}\right] \boldsymbol{z}_i < \log|\widetilde{\boldsymbol{\Lambda}}_2| - \log|\widetilde{\boldsymbol{\Lambda}}_1|\right) \tag{24}$$

where the quadratic form

$$\frac{1}{G}\sum_{i=1}^{G} \boldsymbol{z}_i^H \left[\boldsymbol{I} - \widetilde{\boldsymbol{\Lambda}}_1 \widetilde{\boldsymbol{\Lambda}}_2^{-1}\right] \boldsymbol{z}_i = \frac{1}{G}\sum_{i=1}^{G}\sum_{n=1}^{N} |z_i[n]|^2 \left(\frac{\lambda_2[n] - \lambda_1[n]}{\lambda_2[n] + 1/n_G}\right) \tag{25}$$

is a weighted sum of NG χ_2^2 random variables ($\{|z_i[n]|^2\}$) whose density and distribution functions can be analytically computed but are tedious [7]. Similarly, under H_2

$$PD_2 = P\left(\frac{1}{G}\sum_{i=1}^{G} \boldsymbol{z}_i^H \left[\boldsymbol{I} - \widetilde{\boldsymbol{\Lambda}}_2 \widetilde{\boldsymbol{\Lambda}}_1^{-1}\right] \boldsymbol{z}_i < \log|\widetilde{\boldsymbol{\Lambda}}_1| - \log|\widetilde{\boldsymbol{\Lambda}}_2|\right). \tag{26}$$

We note that the PD can be computed in closed form for $M = 2$, as indicated above, but we do not provide the explicit expression since it is rather tedious.

3.3 Decision-Fusion Classifier

The test statistic for the decision-fusion classifier takes the form

$$l_{df,j}(\boldsymbol{x}_1, \cdots, \boldsymbol{x}_K) = \log|\widehat{\boldsymbol{\Lambda}}_j| + \frac{1}{K}\sum_{i=1}^{K} \boldsymbol{x}_i^H \widehat{\boldsymbol{\Lambda}}_j^{-1} \boldsymbol{x}_i \quad , \quad \widehat{\boldsymbol{\Lambda}}_j = \boldsymbol{\Lambda}_j + \boldsymbol{I}. \tag{27}$$

The quadratic form in the test statistic can be expanded as

$$\begin{aligned}\frac{1}{K}\sum_{i=1}^{K}\boldsymbol{x}_i^H\widehat{\boldsymbol{\Lambda}}_j^{-1}\boldsymbol{x}_i &= \frac{1}{Gn_G}\sum_{i=1}^{G}\sum_{k=1}^{n_G}(\boldsymbol{s}_i+\boldsymbol{n}_{(i-1)G+k})^H\widehat{\boldsymbol{\Lambda}}_j^{-1}(\boldsymbol{s}_i+\boldsymbol{n}_{(i-1)G+k}) \\ &= \frac{1}{G}\sum_{i=1}^{G}\left[\boldsymbol{s}_i^H\widehat{\boldsymbol{\Lambda}}_j^{-1}\boldsymbol{s}_i+2\mathrm{Re}\left[\boldsymbol{s}_i^H\widehat{\boldsymbol{\Lambda}}_j^{-1}\boldsymbol{w}_i\right]\right. \\ &\quad \left.+\frac{1}{n_G}\sum_{k=1}^{n_G}\boldsymbol{n}_{(i-1)G+k}^H\widehat{\boldsymbol{\Lambda}}_j^{-1}\boldsymbol{n}_{(i-1)G+k}\right] \end{aligned} \tag{28}$$

where $\boldsymbol{s}_i$ and $\boldsymbol{w}_i$ are defined in (8) and (21). The density for the above quadratic form can be computed exactly but here we provide a simple approximation that yields fairly accurate (but conservative) PD estimates and relates $l_{df,j}$ to $l_{opt,j}$. We make two approximations. First, we replace the $\boldsymbol{w}_i \sim \mathcal{CN}(\boldsymbol{0}, \boldsymbol{I}/n_G)$ in (28) with $\boldsymbol{w}_i \sim \mathcal{CN}(\boldsymbol{0}, \boldsymbol{I})$. Second, we replace $\frac{1}{n_G}\sum_{k=1}^{n_G}\boldsymbol{n}_{(i-1)G+k}^H\widehat{\boldsymbol{\Lambda}}_j^{-1}\boldsymbol{n}_{(i-1)G+k}$ with $\boldsymbol{w}_i\widehat{\boldsymbol{\Lambda}}_j^{-1}\boldsymbol{w}_i$, $\boldsymbol{w}_i \sim \mathcal{CN}(\boldsymbol{0}, \boldsymbol{I})$. With the above approximations we have

$$\frac{1}{K}\sum_{i=1}^{K}\boldsymbol{x}_i^H\widehat{\boldsymbol{\Lambda}}_j^{-1}\boldsymbol{x}_i \approx \frac{1}{G}\sum_{i=1}^{G}\widehat{\boldsymbol{y}}_i^H\widehat{\boldsymbol{\Lambda}}_j^{-1}\widehat{\boldsymbol{y}}_i \tag{29}$$

where $\widehat{\boldsymbol{y}}_i$ are i.i.d. $\mathcal{CN}(\boldsymbol{0}, \widehat{\boldsymbol{\Lambda}}_j)$ under H_j. Thus, the PD_1 and PD_2 for the decision fusion classifier can be approximated by those of the optimal classifier given in (24) and (26) by replacing $\widetilde{\boldsymbol{\Lambda}}_j$ with $\widehat{\boldsymbol{\Lambda}}_j$. In particular, the quadratic form for PD_1 is given by

$$\frac{1}{G}\sum_{i=1}^{G}\boldsymbol{z}_i^H\left[\boldsymbol{I}-\widehat{\boldsymbol{\Lambda}}_1\widehat{\boldsymbol{\Lambda}}_2^{-1}\right]\boldsymbol{z}_i = \frac{1}{G}\sum_{i=1}^{G}\sum_{n=1}^{N}|z_i[n]|^2\left(\frac{\lambda_2[n]-\lambda_1[n]}{\lambda_2[n]+1}\right) \tag{30}$$

which is a weighted sum of NG χ_2^2 random variables ($\{|z_i[n]|^2\}$), as for C_{opt}. However, the weights are different and essentially amount to a loss in SNR by a factor of n_G compared to C_{opt} since C_{df} does not do signal averaging within each group. The above conservative analysis shows that C_{df} fully exploits the independent observations across different groups, as C_{opt}, but incurs an effective loss in SNR compared to C_{opt}.

3.4 Data-Averaging Classifier

The test statistic for the data-averaging classifier takes the form

$$l_{da,j}(\boldsymbol{y}_{da}) = \log|\check{\boldsymbol{\Lambda}}_j| + \boldsymbol{y}_{da}^H\check{\boldsymbol{\Lambda}}_j^{-1}\boldsymbol{y}_{da} \quad , \quad \check{\boldsymbol{\Lambda}}_j = \boldsymbol{\Lambda}_j + \boldsymbol{I}/K \tag{31}$$

where $\boldsymbol{y}_{da} \sim \mathcal{CN}(\boldsymbol{0}, \check{\boldsymbol{\Lambda}}_j)$, $\check{\boldsymbol{\Lambda}}_j = \boldsymbol{\Lambda}_j/G + \boldsymbol{I}/K$ under H_j. Thus, $\boldsymbol{y}_{da}$ can be represented as $\boldsymbol{y}_{da} = \check{\boldsymbol{\Lambda}}_j^{1/2}\boldsymbol{z}$ where $\boldsymbol{z} \sim \mathcal{CN}(\boldsymbol{0}, \boldsymbol{I})$. Proceeding similarly as above, it

can be shown that

$$PD_1 = P\left(\boldsymbol{z}^H \check{\boldsymbol{\Lambda}}_1 \left[\check{\boldsymbol{\Lambda}}_1^{-1} - \check{\boldsymbol{\Lambda}}_2^{-1}\right] \boldsymbol{z} < \log|\check{\boldsymbol{\Lambda}}_2| - \log|\check{\boldsymbol{\Lambda}}_1|\right) \tag{32}$$

where the quadratic form can be expressed as

$$\boldsymbol{z}^H \check{\boldsymbol{\Lambda}}_1 \left[\check{\boldsymbol{\Lambda}}_1^{-1} - \check{\boldsymbol{\Lambda}}_2^{-1}\right] \boldsymbol{z} = \sum_{n=1}^{N} |z[n]|^2 \left(\frac{\lambda_1[n]}{G} + \frac{1}{K}\right)\left(\frac{\lambda_2[n] - \lambda_1[n]}{(\lambda_1[n] + 1/K)(\lambda_2[n] + 1/K)}\right) \tag{33}$$

which is a weighted sum of N χ_2^2 random variables ($\{|z[n]|^2\}$). Similarly,

$$PD_1 = P\left(\boldsymbol{z}^H \check{\boldsymbol{\Lambda}}_2 \left[\check{\boldsymbol{\Lambda}}_2^{-1} - \check{\boldsymbol{\Lambda}}_1^{-1}\right] \boldsymbol{z} < \log|\check{\boldsymbol{\Lambda}}_1| - \log|\check{\boldsymbol{\Lambda}}_2|\right). \tag{34}$$

The density and distribution function of the quadratic form in (33) can be computed in closed form [7] and thus the PD of the data-averaging classifier can also be computed in closed form.

The data-averaging classifier provides maximum immunity against noise by averaging over all measurements. However, it does exploit the independent signal component in different groups to reduce the inherent variations in the signal. Thus, in the limit of large number of uncorrelated measurements, we expect both C_{opt} and C_{df} to exhibit improved performance (perfect classification under certain conditions), but the performance of C_{da} will always be limited.

3.5 Asymptotic Performance

We now analyze classifier performance in the limit of large G (and K) for fixed n_G. In this analysis, we consider arbitrary $M > 2$. According to the analysis above, the only effect of n_G is to alter the effective SNR in the case of C_{opt} and C_{df}. First, consider the optimal classifier. Note that

$$l_{opt,j}(\boldsymbol{y}_1, \cdots, \boldsymbol{y}_G) = -\log p_j(\boldsymbol{y}_1, \cdots, \boldsymbol{y}_G)/G = -\frac{1}{G}\sum_{i=1}^{G} \log p_j(\boldsymbol{y}_i) \tag{35}$$

since $\boldsymbol{y}_i$ are i.i.d. $\mathcal{CN}(\boldsymbol{0}, \widetilde{\boldsymbol{\Lambda}}_m)$ under H_m. Thus, under H_m, it is well-known that by the law of large numbers [8]

$$\lim_{G\to\infty} l_{opt,j}(\boldsymbol{y}_1, \cdots, \boldsymbol{y}_G) = -\mathrm{E}_m[\log p_j(\boldsymbol{Y})] = D(p_m \| p_j) + h_m(\boldsymbol{Y}) \tag{36}$$

where $\mathrm{E}_m[\cdot]$ denotes expectation under H_m, $D(p_m\|p_j)$ is the Kullback-Leibler distance between p_j and p_m [8]

$$D(p_m\|p_j) = \mathrm{E}_m\left[\log(p_m(\boldsymbol{Y})/p_j(\boldsymbol{Y}))\right] = \log\left(|\widetilde{\boldsymbol{\Lambda}}_j|/|\widetilde{\boldsymbol{\Lambda}}_m|\right) + \mathrm{tr}\left(\widetilde{\boldsymbol{\Lambda}}_j^{-1}\widetilde{\boldsymbol{\Lambda}}_m - \boldsymbol{I}\right) \tag{37}$$

and $h_m(\boldsymbol{Y})$ is the differential entropy (in bits) of $\boldsymbol{y}_i$ under H_m [8]

$$h_m(\boldsymbol{Y}) = -\mathrm{E}_m[\log p_m(\boldsymbol{Y})] = \log\left((\pi e)^N |\widetilde{\boldsymbol{\Lambda}}_m|\right). \tag{38}$$

From (36), we note that under H_m the different test statistics (for $j = 1, \cdots, M$) differ only in the term $D(p_m \| p_j) \geq 0$ which is identically zero for $j = m$. Thus, perfect classification ($PD = 1$, $PFA = 0$) is attained in the limit of large G if

$$D(p_m \| p_j) > 0 \ \forall j, m, j \neq m \tag{39}$$

which would be true in general for any given SNR (and any fixed n_G).

Now consider the decision-fusion classifier. Recall from (27) and (29) that the test statistics can be conservatively approximated as

$$l_{df,j}(\widehat{\boldsymbol{y}}_1, \cdots, \widehat{\boldsymbol{y}}_G) \approx -\log \widehat{p}_j(\widehat{\boldsymbol{y}}_1, \cdots, \widehat{\boldsymbol{y}}_G)/G = \frac{1}{G}\sum_{i=1}^{G} -\log \widehat{p}_j(\widehat{\boldsymbol{y}}_i) \tag{40}$$

since $\widehat{\boldsymbol{y}}_i$ are i.i.d. $\mathcal{CN}(\boldsymbol{0}, \widehat{\boldsymbol{\Lambda}}_m)$ under H_m and $\widehat{p}_j$ denotes the density of $\mathcal{CN}(\boldsymbol{0}, \widehat{\boldsymbol{\Lambda}}_j)$. Thus, in the limit of large G (under H_m)

$$\lim_{G\to\infty} l_{df,j}(\widehat{\boldsymbol{y}}_1, \cdots, \widehat{\boldsymbol{y}}_G) = -\mathrm{E}_m[\log \widehat{p}_j(\widehat{\boldsymbol{Y}})] = D(\widehat{p}_m \| \widehat{p}_j) + h_m(\widehat{\boldsymbol{Y}}) \tag{41}$$

where $D(\widehat{p}_m \| \widehat{p}_j)$ and $h_m(\widehat{\boldsymbol{Y}})$ are defined similar to (37) and (38). Consequently, in the limit of large G we expect perfect classification if

$$D(\widehat{p}_m \| \widehat{p}_j) > 0 \ \forall j, m, j \neq m \tag{42}$$

which would also be true in general for any given SNR (and any fixed n_G).

Finally, consider the data-averaging classifier whose test statistics are given in (31) where $\boldsymbol{y}_{da} \sim \mathcal{CN}(\boldsymbol{0}, \breve{\boldsymbol{\Lambda}}_m)$ under H_m. Recall that $\breve{\boldsymbol{\Lambda}}_j = \boldsymbol{\Lambda}_j + \boldsymbol{I}/K$ and $\breve{\boldsymbol{\Lambda}}_j = \boldsymbol{\Lambda}_j/G + \boldsymbol{I}/K$. As $G(K) \to \infty$, $\breve{\boldsymbol{\Lambda}}_j \to \boldsymbol{\Lambda}_j$ and $\breve{\boldsymbol{\Lambda}}_j \to \boldsymbol{0}$. Consequently,

$$\lim_{G\to\infty} l_{da,j}(\boldsymbol{y}_{da}) = \log|\boldsymbol{\Lambda}_j| \tag{43}$$

independent of the true underlying hypothesis. Thus, in the limit of large G (K), the data-averaging classifier assigns every event to the class with the smallest value of $\log|\boldsymbol{\Lambda}_j|$ and results in worst performance ($PD = PFA = 1/M$).

3.6 Entropy Comparison Between Data and Decision Fusion

The above analysis indicates that C_{df} approximates the performance of C_{opt} except for an SNR loss depending on the fraction of correlated measurements n_G. The numerical results in the next section confirm the analysis. However, the attractiveness of C_{df} is also implicitly based on the assumption that communicating the likelihoods from the K nodes to the manager node puts a smaller communication burden on the network compared to communicating the N-dimensional feature vectors in the case of C_{opt}.

Recall from (13) that in C_{df} the M quadratic forms $\{\boldsymbol{x}_i^H(\boldsymbol{\Sigma}_j+\boldsymbol{I})^{-1}\boldsymbol{x}_i\ ,\ j=1,\cdots,M\}$ are communicated from the i-th node to the manager node for $i=1,\cdots,K$. In C_{opt}, on the other hand, the N-dimensional vectors $\boldsymbol{x}_i$ are communicated from the K nodes to the manager node. Thus, we need to compare the cost of communicating M quadratic forms (scalars) to that of communicating an N-dimensional Gaussian vector from each node to the manager node. We compare the communication cost in terms of differential entropy [8].[1]

The differential entropy of $\boldsymbol{x}\sim\mathcal{CN}(\boldsymbol{0},\boldsymbol{\Sigma}_m+\boldsymbol{I})$ is

$$h_m(\boldsymbol{X})=-\mathrm{E}[\log p_m(\boldsymbol{X})]=\log\left((\pi e)^N|\boldsymbol{\Sigma}_m+\boldsymbol{I}|\right)=\log\left((\pi e)^N|\boldsymbol{\Lambda}_m+\boldsymbol{I}|\right) \quad (44)$$

and quantifies the information content of any $\boldsymbol{x}_i$ from m-th class.

Now consider the differential entropy of the quadratic forms used by C_{df}. Let q_{jm} denote the quadratic form associated with $l_{df,j}$ under H_m

$$q_{jm}=\boldsymbol{x}^H(\boldsymbol{\Sigma}_j+\boldsymbol{I})^{-1}\boldsymbol{x}=\boldsymbol{z}^H\boldsymbol{\Lambda}_{jm}\boldsymbol{z} \quad (45)$$

where $\boldsymbol{x}\sim\mathcal{CN}(\boldsymbol{0},\boldsymbol{\Sigma}_m+\boldsymbol{I})$ and the second equality is based on the eigen-decomposition

$$(\boldsymbol{\Sigma}_m+\boldsymbol{I})^{1/2}(\boldsymbol{\Sigma}_j+\boldsymbol{I})^{-1}(\boldsymbol{\Sigma}_m+\boldsymbol{I})^{1/2}=\boldsymbol{U}\boldsymbol{\Lambda}_{jm}\boldsymbol{U}^H \quad (46)$$

which uses the representation $\boldsymbol{x}=(\boldsymbol{\Sigma}_m+\boldsymbol{I})^{1/2}\boldsymbol{z}$, $\boldsymbol{z}\sim\mathcal{CN}(\boldsymbol{0},\boldsymbol{I})$. Note that under the simultaneously diagonalizable signal model we have

$$\boldsymbol{\Lambda}_{jm}=\widehat{\boldsymbol{\Lambda}}_m\widehat{\boldsymbol{\Lambda}}_j^{-1}\ ,\ \widehat{\boldsymbol{\Lambda}}_j=\boldsymbol{\Lambda}_j+\boldsymbol{I}. \quad (47)$$

We want to compute the entropy of the quadratic form random variable Q_{jm} for all j,m. We first compute the worst case (highest) entropy by assuming that Q_{jm} is Gaussian. Using the fact that $\boldsymbol{x}$ is Gaussian, it can be readily shown that

$$\mathrm{E}[Q_{jm}]=\mathrm{E}_m[\boldsymbol{x}^H(\boldsymbol{\Sigma}_j+\boldsymbol{I})^{-1}\boldsymbol{x}]=\mathrm{tr}\left(\boldsymbol{\Lambda}_{jm}\right)\ ,\ \mathrm{var}(Q_{jm})=\mathrm{tr}\left(\boldsymbol{\Lambda}_{jm}^2\right). \quad (48)$$

Thus, the worst-case entropy of Q_{jm} is given by [8]

$$h(Q_{jm})=\frac{1}{2}\log\left(2\pi e\ \mathrm{tr}\left(\boldsymbol{\Lambda}_{jm}^2\right)\right). \quad (49)$$

Note from (47) that $\boldsymbol{\Lambda}_{jm}=\boldsymbol{I}$ for $j=m$. Thus, $h(Q_{jj})$ is the same for all j. For $j=m$, the true entropy can also be easily computed since $q_{jj}=\|\boldsymbol{z}\|^2$ from (45). Now, $q=\|\boldsymbol{z}\|^2\sim\chi^2_{2N}$ with density given by [7]

$$p_Q(q)=\frac{1}{(N-1)!}q^{N-1}e^{-q}\ ,q\geq 0 \quad (50)$$

[1] We note differential entropy can be a bit misleading since it can be negative. However, a comparison of the difference in differential entropies is still valid – a quantity with higher entropy would require more bits to encode. A more intuitive interpretation of differential entropy is based on the fact that the entropy of an n-bit quantization of continuous random variable X is approximately $h(X)+n$ [8].

and thus

$$h(Q) = -\mathrm{E}[\log p_Q(Q)] = \log(N-1)! - N - (N-1)\int_0^\infty p_Q(q)\log(q)dq \ . \quad (51)$$

We note that the true entropy of q_{jm}, for $j \neq m$, can also be computed in closed-form but it is a bit more involved. Furthermore, as our numerical results indicate, $h(Q)$ is a good estimate for the entropy of q_{jm} for all j, m.

4 Simulation Results Based on Real Data

We now present numerical results based on real data collected in the SensIT program. We consider the problem of classifying a single vehicle. We consider $M = 2$ classes: Amphibious Assault Vehicle (AAV; tracked vehicle) and Dragon Wagon (DW; wheeled vehicle). We simulated $N = 25$ dimensional acoustic measurements from $K = Gn_G = 10$ nodes according to the model in Section 2.3. The eigenvalues (PSD samples) for the two vehicles were estimated from experimental data. The measurements at different nodes were generated using (21). The PD and PFA were estimated using Monte Carlo simulation over 5000 independent events. For $\boldsymbol{x} \sim \mathcal{CN}(\boldsymbol{0}, \boldsymbol{\Lambda} + \boldsymbol{I})$, $\mathrm{SNR} = \mathrm{tr}(\boldsymbol{\Lambda})/\mathrm{tr}(\boldsymbol{I})$.

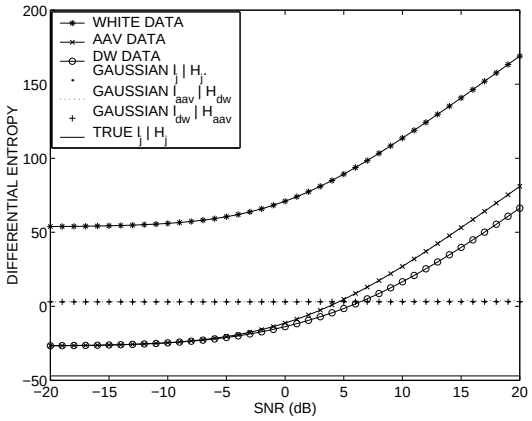

Fig. 1. Comparison between differential entropies of $N = 25$ dimensional Gaussian vectors used by l_{opt} and those of the quadratic forms used by l_{df}. The entropies for a white vector and two correlated vectors (AAV and DW) are plotted. The worst-case entropies for quadratic forms, assuming Gaussian statistics, are plotted. The true entropy of the quadratic forms, under the correct hypothesis, is also plotted. The entropy gains of decision fusion over data fusion are evident.

Figure 1 compares the differential entropy of Gaussian data in (44) with that of the quadratic forms in (49) and (51). The entropies for three data vectors are plotted: white data (maximum entropy), AAV data, and DW data. The

worst-case entropy in (49) of the quadratic forms used by C_{df} are also plotted for all j, m (they are nearly identical). It can be seen that for SNR above 5dB, the worst-case entropy of Q_{jm} is lower than that of data. The true entropy of Q_{jj}, given in (51), is also plotted for comparison. The true entropy of Q_{jj} is seen to be substantially lower compared to that of data for the entire SNR range considered. This indicates the significant potential gains of C_{df} over C_{opt} in terms of the communication burden.

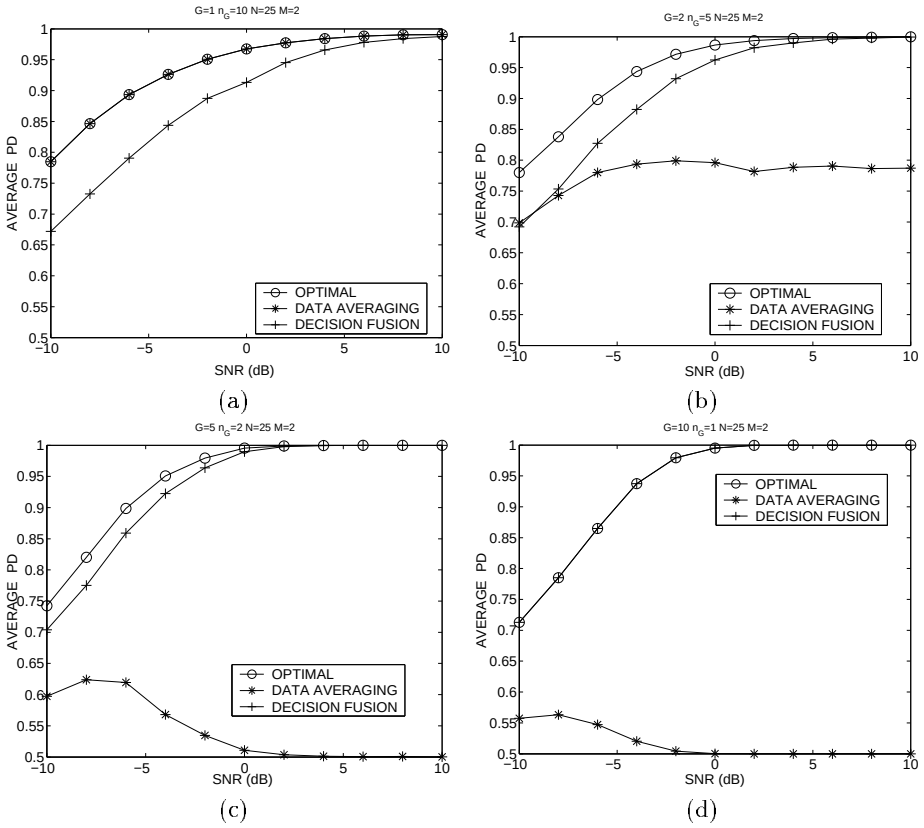

Fig. 2. PD of the three classifiers versus SNR. (a) $K = n_G = 10$ (perfectly correlated measurements). (b) $G = 2$ and $n_G = 5$. (c) $G = 5$ and $n_G = 2$. (d) $K = G = 10$ (independent measurements).

Figure 2 plots the PD as a function of SNR for the three classifiers for $K = 10$ and different combinations of G and n_G. The PFA is simply given by $1 - PD$ for $M = 2$. As expected, C_{opt} and C_{da} perform identically for $K = n_G$ (perfectly correlated case; Figure 2(a)), whereas C_{opt} and C_{df} perform identically for $K = G$ (independent case; Figure 2(d)). Note that C_{df} incurs a small loss in performance in the perfectly correlated (worst) case which diminishes at high SNRs. The performance loss in C_{da} in the independent (worst) case is very sig-

nificant and does not improve with SNR. This is consistent with our analysis. At high SNR, all events are classified as DW by C_{da} since $\log|\boldsymbol{\Lambda}_{DW}| < \log|\boldsymbol{\Lambda}_{AAV}|$ due to the peakier eigenvalue distribution for DW, as evident from Figure 3(a). Figure 3(b) compares the PD of the three classifiers for an intermediate case ($G = n_G = 2$) with $K = 4$, $N = 15$-dimensional measurements. Analytically computed PD for C_{opt} and C_{da} and the conservative approximation for PD of C_{df} are also plotted and agree well with the simulation results.

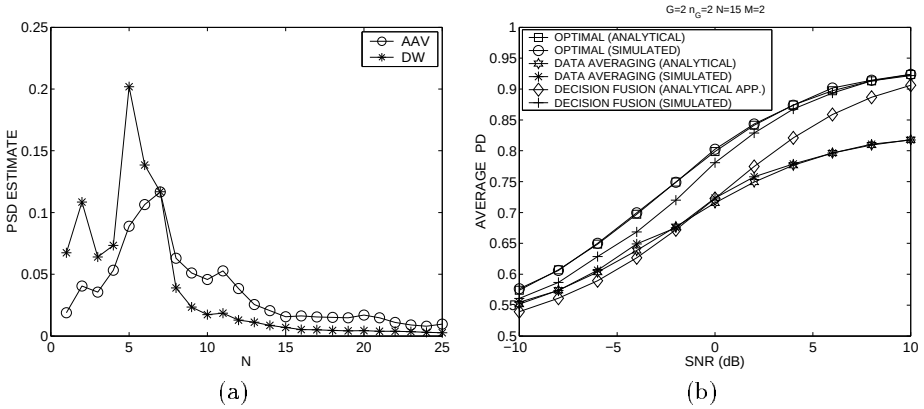

Fig. 3. (a) Covariance matrix eigenvalues (PSD estimates) for AAV and DW. (b) Comparison of simulated and analytical PD for $K = 4$, $G = 2$, $n_G = 2$ and $N = 15$.

Figure 4 plots the PD for the three classifiers as function of G ($K = 10$) for two different SNRs. It is evident that C_{df} closely approximates C_{opt} whereas C_{da} incurs a large loss when $K \neq n_G$. It is worth noting that for SNR=-5dB, the performance of C_{opt} and C_{df} first improves slightly with G and then gets worse again. This is consistent with the observation, in non-coherent communication over fading channels, that there is an optimal level of diversity (G) for a given SNR – increasing G beyond that level results in a loss in performance [7].

5 Conclusions

We have taken a first step in addressing the problem of how much information should be exchanged between nodes for distributed decision making in sensor networks. Our analysis is based on modeling the source signal as a stationary Gaussian process. In general, measurements from multiple nodes will provide a mixture of correlated and uncorrelated information about the source signal. The optimal classifier exploits the correlated measurements to improve the SNR and the independent measurements to stablize the inherent statistical variability in the signal. Both effects are important for improving classifier performance. However, for stochastic signals, the fusion of independent measurements is most significant. In this context, our results demonstrate that the simple sub-optimal decision-fusion classifier, that treats all measurements as independent, is not only

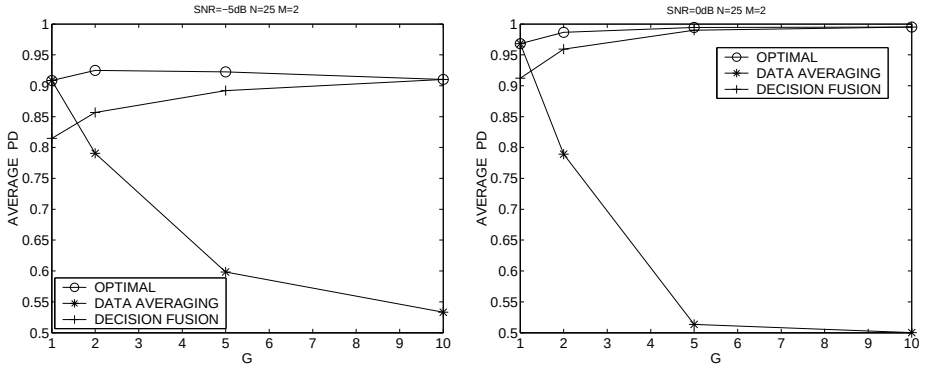

Fig. 4. Comparison of PD of the three classifiers for varying values of G ($K = 10$). (a) SNR = -5 dB. (b) SNR = 0 dB.

an attractive choice given the computational and communication constraints in a sensor network, but is also a robust choice from a decision theoretic viewpoint. The decision-fusion classifier fully exploits the independent measurements and only incurs an effective SNR loss compared to the optimal classifier depending on the fraction of correlated measurements. However, if the source signal exhibits a non-zero mean or fewer degrees of freedom (lower-rank covariance matrix), data averaging to improve SNR might become more important. We note that exploiting a non-zero mean is difficult in practice due to various sources of measurement error. Directions for future research include hard decision fusion, quantized measurements, and multiple-target classification.

References

[1] D. Estrin, L. Girod, G. Pottie, and M. Srivastava, "Instrumenting the world with wireless sensor network," in *Proc. IEEE Int. Conf. on Acoust., Speech and Signal Proc. – ICASSP'01*, 2001, pp. 2675–2678.

[2] "Special issue on collaborative signal and information processing in microsensor networks," in *IEEE Signal Processing Magazine*. (S. Kumar and F. Zhao and D. Shepherd (eds.), March 2002.

[3] D. Li, K. Wong, Y. Hu, and A. Sayeed, "Detection, classification, tracking of targets in micro-sensor networks," in *IEEE Signal Processing Magazine*, March 2002, pp. 17–29.

[4] R. Duda, P. Hart, and D. Stork, *Pattern Classification*, Wiley, 2nd edition, 2001.

[5] J. Kittler, M. Hatef, R. Duin, and J. Matas, "On combining classifiers," *IEEE Trans. Pattern Anal. Machine Intelligence*, vol. 20, no. 3, pp. 226–238, Mar. 1998.

[6] Robert M. Gray, "On the asymptotic eigenvalue distribution of toeplitz matrices," *IEEE Trans. Inform. Th.*, vol. 18, no. 6, pp. 725–730, Nov. 1972.

[7] J. G. Proakis, *Digitial Communications*, McGraw Hill, New York, 3rd edition, 1995.

[8] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, Wiley, 1991.