In-Situ Calibration of Sensor Networks for Distributed Detection Applications

Gernot Fabeck [1], Rudolf Mathar [2]
[1,2] *Institute of Theoretical Information Technology, RWTH Aachen University*
Aachen, Germany, {fabeck,mathar}@ti.rwth-aachen.de

Abstract

In this paper, we present algorithms for in-situ calibration of sensor networks for distributed detection in the parallel fusion architecture. The wireless sensors act as local detectors and transmit preliminary detection results to an access point or fusion center for decision combining. In order to implement an optimal fusion center, both the performance parameters of each local detector (i.e., its probability of false alarm and probability of miss) as well as the wireless channel conditions must be known. However, in real-world applications these statistics may be unknown or vary in time. In our approach, the fusion center receives a collection of labeled samples from the sensor nodes after deployment of the network and calibrates the impact of individual sensors on the final detection result. In the case that local sensor decisions are independent, we employ maximum likelihood parameter estimation techniques, whereas in the case of arbitrarily correlated sensor outputs, we use the method of kernel smoothing. The obtained fusion rules are both asymptotically optimal and show good performance for finite sample sizes.

1. INTRODUCTION

Important applications of wireless sensor networks include the detection of targets in the monitored environment for surveillance purposes, e.g., as a prerequisite for tracking [1]. The wireless sensor nodes typically operate on limited energy budgets and are consequently subject to communication constraints, resulting in a finite number of bits each sensor node can transmit to the data sink before it runs out of power. In order to extend sensor network lifetime, preprocessing of measured raw data at the sensors and transmission of summary messages is recommended. In the parallel fusion architecture, the sensor nodes process their observations independently and make preliminary decisions about the state of the observed environment. The sensors transmit the local decisions to a fusion center that combines the received messages and computes the final decision. In the case that the state of the environment is indicated by a binary variable, the problem of designing the local sensor compression rules and the fusion rule with respect to an overall performance criterion is called the problem of *distributed detection*.

Distributed detection is a well-developed field of research that traces back to the early work of Tenney and Sandell [2]. Over the last two and a half decades, the problem of distributed detection has been analyzed under a plethora of different aspects concerning performance criteria, network topologies, and communication schemes [3], [4]. Despite the host of investigated scenarios, the majority of the literature relies on strong assumptions with respect to the underlying statistical model, e.g., explicit knowledge of the performance parameters of the local detectors as well as the correlations between local detector outputs. In the area of wireless sensor networks, these assumptions may become infeasible because one encounters random deployment of sensor nodes and heterogeneous sensing environments. In such scenarios, detailed knowledge of the underlying statistical model may not be available for system design. To tackle these difficulties, in-situ calibration approaches are recommended which facilitate configuration of sensor networks for detection applications *on the spot*.

A suitable framework for the design of in-situ calibration algorithms for distributed detection in sensor networks is the field of *supervised learning* [5]. Supervised learning refers to learning from labeled samples of the underlying unknown probability distribution. Instead of determining the system in advance relying on prior knowledge, the deployed sensor network undergoes a learning phase in which unknown quantities are learnt from a recorded training set. In this paper, we introduce algorithms for in-situ calibration of sensor networks for distributed detection in the parallel fusion architecture based on supervised learning. We distinguish two cases, depending on whether local sensor detection results are correlated or not. Under the assumption of conditionally independent sensor decisions, we employ maximum likelihood parameter estimation to implement a fusion center which asymptotically achieves minimum probability of error. In the case of arbitrarily correlated sensor decisions, we use the method of kernel smoothing. Using suitable discrete kernel functions and computing the right smoothing parameters, this method also results in fusion rules which asymptotically provide optimal detection performance for any distribution.

The remainder of this paper is organized as follows. In Section 2, the problem of distributed detection in the parallel fusion network with noisy communication links is stated. In Section 3, we formulate the in-situ calibration algorithm for conditionally independent local sensor decisions. In-situ calibration of the sensor network in the general case of correlated sensor decisions is discussed in Section 4. In Section 5, we present numerical results obtained by Monte Carlo simulation and we conclude in Section 6.

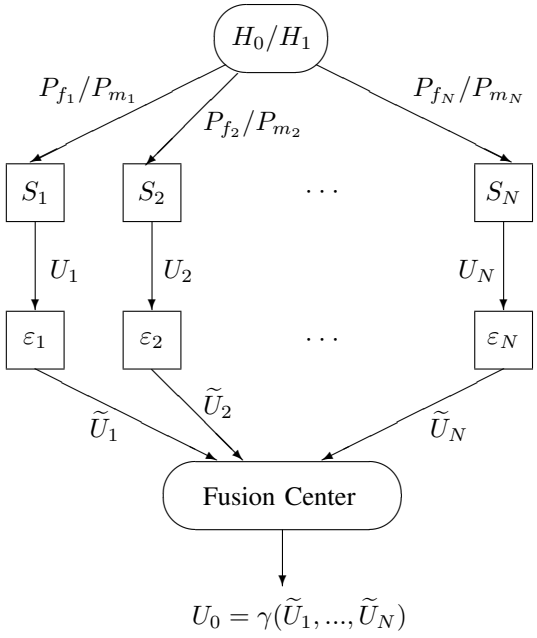

Fig. 1: Parallel fusion network with noisy communication links.

2. PARALLEL FUSION NETWORK WITH NOISY COMMUNICATION LINKS

The problem of distributed detection in the parallel fusion network with noisy communication links is as follows (see Fig. 1). We consider a binary hypothesis testing problem with hypotheses H_0 and H_1 indicating the state of the observed environment, e.g., absence or presence of a target. The actual state of the environment is described by a binary-valued random variable $Y \in \{0, 1\}$ and associated prior probabilities

$$\begin{aligned} \pi_0 &= P(H_0) = P(Y = 0), \\ \pi_1 &= P(H_1) = P(Y = 1). \end{aligned} \tag{1}$$

In order to infer the true state of nature, a network of N sensors $S_1, ..., S_N$ collect measurement data generated according to either H_0 or H_1, the two hypotheses under test. Each sensor processes its observation independently and makes a preliminary decision about the true hypothesis before sending it to a fusion center. In the case that every wireless sensor is allowed to transmit one bit per observation, the sensor decisions are binary-valued random variables $U_j \in \{0, 1\}$, $j = 1, ..., N$. The resulting detection error probabilities for each sensor are given by the local probability of false alarm P_{f_j} and the local probability of miss P_{m_j} according to

$$P_{f_j} = P(U_j = 1|H_0), \tag{2}$$

$$P_{m_j} = P(U_j = 0|H_1), \tag{3}$$

for $j = 1, ..., N$. Upon local detection, the sensor nodes transmit an array of local decisions

$$\boldsymbol{U} = (U_1, ..., U_N) \in \{0, 1\}^N \tag{4}$$

to the fusion center in order to perform decision combining. The communication links between the wireless sensors and the fusion center may be subject to noise. Due to noisy communication channels, the fusion center receives an array of potentially distorted decisions

$$\widetilde{\boldsymbol{U}} = (\widetilde{U}_1, ..., \widetilde{U}_N) \in \{0, 1\}^N. \tag{5}$$

We follow an approach suggested by Ferrari and Pagliari [6] and model the communication link between sensor S_j and the fusion center by a binary symmetric channel (BSC) with bit-error probability ε_j, i.e.

$$\varepsilon_j = P(\widetilde{U}_j = 1|U_j = 0) = P(\widetilde{U}_j = 0|U_j = 1) \tag{6}$$

for $j = 1, ..., N$. The received decisions are combined to yield the final decision $U_0 = \gamma(\widetilde{U}_1, ..., \widetilde{U}_N)$, where the *fusion rule* γ is a binary-valued mapping

$$\gamma : \{0, 1\}^N \to \{0, 1\}. \tag{7}$$

The sensor network detection performance is measured in terms of the global *probability of error*

$$\begin{aligned} P_e &= P(\gamma(\widetilde{\boldsymbol{U}}) \neq Y) \\ &= \pi_0 P_f + \pi_1 P_m \end{aligned} \tag{8}$$

which can be written as a weighted sum of the global probability of false alarm $P_f = P(U_0 = 1|H_0)$ and the corresponding global probability of miss $P_m = P(U_0 = 0|H_1)$. Since the decision fusion problem can be viewed as a hypothesis testing problem with local detection results being the observations, the Bayes optimal fusion rule γ^* takes the form of a *likelihood ratio test*

$$\frac{p(\widetilde{\boldsymbol{u}}|H_1)}{p(\widetilde{\boldsymbol{u}}|H_0)} \underset{0}{\overset{1}{\gtrless}} \frac{\pi_0}{\pi_1}, \tag{9}$$

where $p(\widetilde{\boldsymbol{u}}|H_k) = p(\widetilde{u}_1, ..., \widetilde{u}_N|H_k)$ is the probability mass function of the received decision vector $\widetilde{\boldsymbol{U}} = (\widetilde{U}_1, ..., \widetilde{U}_N)$ under hypothesis H_k, $k = 0, 1$. The minimum probability of error associated with the optimal fusion rule (9) is given by the *Bayes risk*

$$P_e^* = P(\gamma^*(\widetilde{\boldsymbol{U}}) \neq Y). \tag{10}$$

3. IN-SITU CALIBRATION FOR INDEPENDENT SENSORS VIA PARAMETER ESTIMATION

In this section we will introduce in-situ calibration of sensor networks in the special case that the local detection results at the sensors are conditionally independent given the underlying hypothesis. Using maximum likelihood parameter estimation, we obtain Bayes risk consistent fusion rules which asymptotically achieve minimum detection error.

A. Distribution of received detection results

Under the assumption of conditional independence, the probability mass functions of the decision vector $\boldsymbol{U} = (U_1, ..., U_N)$

factorize and take the form

$$p(u_1,...,u_N|H_0) = \prod_{j=1}^{N} P_{f_j}^{u_j}(1-P_{f_j})^{1-u_j}, \tag{11}$$

$$p(u_1,...,u_N|H_1) = \prod_{j=1}^{N} P_{m_j}^{1-u_j}(1-P_{m_j})^{u_j}. \tag{12}$$

Since the decision vector $\boldsymbol{U}$ is transmitted over noisy channels, the fusion center receives a potentially distorted vector $\widetilde{\boldsymbol{U}}$. If the noisy channels are modeled by BSCs as formulated in (6), a little computation yields the transformed detection error probabilities

$$\widetilde{P}_{f_j} = P(\widetilde{U}_j = 1|H_0) = P_{f_j} + \varepsilon_j(1-2P_{f_j}), \tag{13}$$

$$\widetilde{P}_{m_j} = P(\widetilde{U}_j = 0|H_1) = P_{m_j} + \varepsilon_j(1-2P_{m_j}). \tag{14}$$

The distribution of the distorted decision vector $\widetilde{\boldsymbol{U}}$ is obtained by plugging expressions (13) and (14) into equations (11) and (12), respectively. This results in the conditional probability mass functions

$$p(\widetilde{u}_1,...,\widetilde{u}_N|H_0) = \prod_{j=1}^{N} \widetilde{P}_{f_j}^{\widetilde{u}_j}(1-\widetilde{P}_{f_j})^{1-\widetilde{u}_j}, \tag{15}$$

$$p(\widetilde{u}_1,...,\widetilde{u}_N|H_1) = \prod_{j=1}^{N} \widetilde{P}_{m_j}^{1-\widetilde{u}_j}(1-\widetilde{P}_{m_j})^{\widetilde{u}_j}, \tag{16}$$

describing the distribution of received detection results.

B. Optimal channel-aware fusion rule

Inserting expressions (15) and (16) into the Bayes optimal fusion rule (9), and after some transformations, the optimal fusion rule γ^* can be implemented by a linear threshold rule

$$\sum_{j=1}^{N} \lambda_j \widetilde{u}_j \underset{0}{\overset{1}{\gtrless}} \vartheta \tag{17}$$

with individual *sensor weights*

$$\lambda_j = \log\left(\frac{(1-\widetilde{P}_{f_j})(1-\widetilde{P}_{m_j})}{\widetilde{P}_{f_j}\widetilde{P}_{m_j}}\right) \tag{18}$$

for $j = 1,...,N$, and a *decision threshold*

$$\vartheta = \log\left(\frac{\pi_0}{\pi_1}\prod_{j=1}^{N}\frac{1-\widetilde{P}_{f_j}}{\widetilde{P}_{m_j}}\right). \tag{19}$$

In order to perform optimal fusion of the received local detection results, the numerical values of the weights (18) and the threshold (19) have to be available at the fusion center. If both the prior probabilities (1), the original sensor detection error probabilities (2) and (3), as well as the bit-error rates of the BSCs (6) would be perfectly known, the sensor weights and the decision threshold could be computed. However, in real-world scenarios knowledge about those quantities may not be available, necessitating in-situ methods for system calibration.

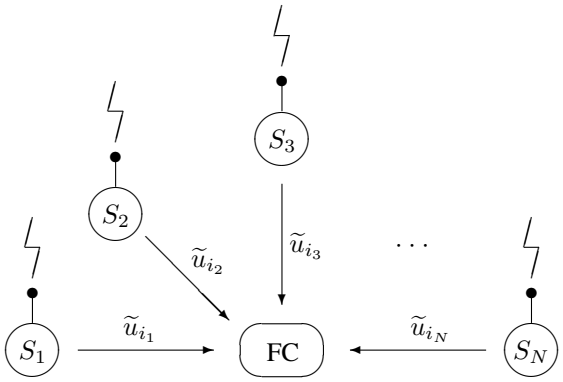

Fig. 2: Sensor nodes $S_1,...,S_N$ transmitting the components of the decision vector $\widetilde{\boldsymbol{u}}_i = (\widetilde{u}_1,...,\widetilde{u}_N)$ to the fusion center (FC).

C. In-situ calibration of the fusion rule

As an operational requirement for in-situ calibration, the fusion center receives a certain number of decision vectors from the sensor nodes after deployment of the network (see Fig. 2). During this phase, each of the decision vectors is assumed to be labeled with the true underlying hypothesis by an external observer. Formally, we assume that a set

$$D_n = \{(\widetilde{\boldsymbol{u}}_i, y_i)\}_{i=1}^{n} \subset \{0,1\}^N \times \{0,1\} \tag{20}$$

of potentially distorted decision vectors $\widetilde{\boldsymbol{u}}_i \in \{0,1\}^N$ labeled with the true underlying hypothesis $y_i \in \{0,1\}$ is available at the fusion center.

In-situ calibration of the fusion rule (17) essentially involves assessing the sensor weights $\lambda_1,...,\lambda_N$ and the decision threshold ϑ by using the empirically obtained set D_n. We take the approach of computing the *maximum likelihood estimates* $\widehat{P}_{f_j}$ and $\widehat{P}_{m_j}$ of $\widetilde{P}_{f_j}$ and $\widetilde{P}_{m_j}$, respectively, which are obtained after a little calculation as

$$\widehat{P}_{f_j} = \frac{\sum_{i=1}^{n}\widetilde{u}_{i_j}\cdot(1-y_i)}{n-\sum_{i=1}^{n}y_i}, \qquad \widehat{P}_{m_j} = \frac{\sum_{i=1}^{n}(1-\widetilde{u}_{i_j})\cdot y_i}{\sum_{i=1}^{n}y_i}, \tag{21}$$

for $j = 1,...,N$, where $\widetilde{u}_{i_j}$ denotes the j-th component of the decision vector $\widetilde{\boldsymbol{u}}_i = (\widetilde{u}_{i_1},...,\widetilde{u}_{i_N})$. We estimate the prior probabilities by

$$\widehat{\pi}_0 = \frac{n-\sum_{i=1}^{n}y_i}{n}, \quad \widehat{\pi}_1 = \frac{\sum_{i=1}^{n}y_i}{n}. \tag{22}$$

We use these expressions to form the *plug-in fusion rule*

$$\sum_{j=1}^{N} \widehat{\lambda}_j \widetilde{u}_j \underset{0}{\overset{1}{\gtrless}} \widehat{\vartheta} \tag{23}$$

by defining

$$\widehat{\lambda}_j = \log\left(\frac{(1-\widehat{P}_{f_j})(1-\widehat{P}_{m_j})}{\widehat{P}_{f_j}\widehat{P}_{m_j}}\right) \tag{24}$$

for $j = 1, ..., N$, and

$$\widehat{\vartheta} = \log\left(\frac{\widehat{\pi}_0}{\widehat{\pi}_1}\prod_{j=1}^{N}\frac{1-\widehat{P}_{f_j}}{\widehat{P}_{m_j}}\right). \quad (25)$$

It can be shown that the resulting plug-in fusion rule (23) is universally consistent, i.e., its probability of error P_e converges to the minimum probability of error P_e^* of the theoretically optimal fusion rule (17) with probability one as the size of the training set D_n tends to infinity [7].

4. In-situ Calibration for correlated sensors via kernel smoothing

If we remove the assumption that local sensor detection results are conditionally independent given the underlying hypothesis, the probability distribution of the received decision vector $\widetilde{\boldsymbol{U}} = (\widetilde{U}_1, ..., \widetilde{U}_N)$ will lose its convenient parametric form as stated in (15) and (16). In general, the joint distribution of a vector of N binary variables is characterized by $2^N - 1$ parameters corresponding to the 2^N different observations that can be obtained. For large N, this results in a prohibitive amount of necessary training data for parameter estimation. In the distributed detection literature, the problem of correlated sensors was tackled by allowing only the presence of special kinds of correlations between the local decisions [8], [9].

In order to cope with arbitrary correlation structures, we use an extension of *kernel smoothing* to multivariate binary spaces which was introduced by Aitchison and Aitken [10]. Doing so, we allow the training data itself to determine the correlation structure of the multivariate binary distribution, paving the way for universally consistent fusion rules.

A. Estimation with discrete kernels

As in the case of independent sensors, we assume that a training set

$$D_n = \{(\widetilde{\boldsymbol{u}}_i, y_i)\}_{i=1}^n \subset \{0,1\}^N \times \{0,1\} \quad (26)$$

of correctly labeled but possibly distorted sensor decision vectors $\widetilde{\boldsymbol{u}}_i \in \{0,1\}^N$ is available at the fusion center. We use the training set D_n in combination with a discrete kernel function $K : \{0,1\}^N \to \mathbb{R}$ to form estimates of the conditional probability mass functions $p(\widetilde{\boldsymbol{u}}|H_k)$ according to

$$\widehat{p}(\widetilde{\boldsymbol{u}}|H_0) = \widehat{p}(\widetilde{\boldsymbol{u}}|D_{n_0}) = \frac{1}{n_0}\sum_{i=1}^{n}\mathbb{I}_{\{y_i=0\}}K(\widetilde{\boldsymbol{u}}|\widetilde{\boldsymbol{u}}_i, \boldsymbol{\lambda}_0), \quad (27)$$

$$\widehat{p}(\widetilde{\boldsymbol{u}}|H_1) = \widehat{p}(\widetilde{\boldsymbol{u}}|D_{n_1}) = \frac{1}{n_1}\sum_{i=1}^{n}\mathbb{I}_{\{y_i=1\}}K(\widetilde{\boldsymbol{u}}|\widetilde{\boldsymbol{u}}_i, \boldsymbol{\lambda}_1), \quad (28)$$

where $D_{n_k} \subset D_n$ is the subset of decision vectors with label k, n_k is the number of decision vectors in the set D_{n_k}, and $\boldsymbol{\lambda}_k = (\lambda_{k_1}, ..., \lambda_{k_N})$ is a vector of smoothing parameters for $k = 0, 1$. The kernel-based estimates (27) and (28) are plugged into the Bayes optimal rule (9), yielding the fusion rule

$$\frac{\widehat{p}(\widetilde{\boldsymbol{u}}|H_1)}{\widehat{p}(\widetilde{\boldsymbol{u}}|H_0)} \underset{0}{\overset{1}{\gtrless}} \frac{\widehat{\pi}_0}{\widehat{\pi}_1}. \quad (29)$$

The estimates $\widehat{\pi}_0$ and $\widehat{\pi}_1$ of the prior probabilities are again chosen according to (22). For the kernel function K, we employ a product of univariate kernels $K_1, ..., K_N$ which take the form

$$K_j(\widetilde{u}_j|\widetilde{u}_{i_j}, \lambda_{k_j}) = \begin{cases} \lambda_{k_j} & \text{if } \widetilde{u}_j = \widetilde{u}_{i_j} \\ 1-\lambda_{k_j} & \text{if } \widetilde{u}_j \neq \widetilde{u}_{i_j} \end{cases}, \quad (30)$$

where the smoothing parameters λ_{k_j} are in the interval $[\frac{1}{2}, 1]$, $j = 1, ..., N$. In particular, $\lambda_{k_j} = \frac{1}{2}$ gives the uniform distribution over $\{0,1\}$ whatever the data and $\lambda_{k_j} = 1$ estimates probabilities by the corresponding relative frequencies. Thus, we obtain the discrete kernel function

$$K(\widetilde{\boldsymbol{u}}|\widetilde{\boldsymbol{u}}_i, \boldsymbol{\lambda}_k) = \prod_{j=1}^{N}\lambda_{k_j}^{1-|\widetilde{u}_j - \widetilde{u}_{i_j}|}(1-\lambda_{k_j})^{|\widetilde{u}_j - \widetilde{u}_{i_j}|}. \quad (31)$$

In the special case of identical smoothing parameters, i.e., $\lambda_{k_j} = \lambda_k$ for $j = 1, ..., N$, the kernel function (31) simplifies to the *Aitchison-Aitken kernel*

$$K(\widetilde{\boldsymbol{u}}|\widetilde{\boldsymbol{u}}_i, \boldsymbol{\lambda}_k) = \lambda_k^{N-d_H(\widetilde{\boldsymbol{u}}, \widetilde{\boldsymbol{u}}_i)}(1-\lambda_k)^{d_H(\widetilde{\boldsymbol{u}}, \widetilde{\boldsymbol{u}}_i)} \quad (32)$$

with $d_H(\widetilde{\boldsymbol{u}}, \widetilde{\boldsymbol{u}}_i)$ the *Hamming distance* between the binary vectors $\widetilde{\boldsymbol{u}}$ and $\widetilde{\boldsymbol{u}}_i$. The disadvantage of the Aitchison-Aitken kernel lies in the fact that no discrimination between sensors is possible, due to the uniformly chosen smoothing parameter.

B. Computing the smoothing parameters

For computing the smoothing parameters, we follow an approach due to Tutz [11] that is based on minimizing the leave-one-out estimator for the probability of error of the fusion rule (29), which can be written as

$$\widehat{P}_e = 1 - \Big(\frac{n_0}{n}\sum_{D_{n_0}} f_0(\widetilde{\boldsymbol{u}})I_0(\widetilde{\boldsymbol{u}}) + \frac{n_1}{n}\sum_{D_{n_1}} f_1(\widetilde{\boldsymbol{u}})I_1(\widetilde{\boldsymbol{u}})\Big). \quad (33)$$

The indicator function $I_0(\widetilde{\boldsymbol{u}})$ is defined by the threshold test

$$\frac{\widehat{p}(\widetilde{\boldsymbol{u}}|D_{n_0} \setminus \{\widetilde{\boldsymbol{u}}\})}{\widehat{p}(\widetilde{\boldsymbol{u}}|D_{n_1})} \underset{0}{\overset{1}{\gtrless}} \frac{\widehat{\pi}_1}{\widehat{\pi}_0} \quad (34)$$

and the indicator function $I_1(\widetilde{\boldsymbol{u}})$ is correspondingly defined by

$$\frac{\widehat{p}(\widetilde{\boldsymbol{u}}|D_{n_1} \setminus \{\widetilde{\boldsymbol{u}}\})}{\widehat{p}(\widetilde{\boldsymbol{u}}|D_{n_0})} \underset{0}{\overset{1}{\gtrless}} \frac{\widehat{\pi}_0}{\widehat{\pi}_1}. \quad (35)$$

The value $f_k(\widetilde{\boldsymbol{u}})$ denotes the relative frequency of the binary vector $\widetilde{\boldsymbol{u}}$ in the sample set D_{n_k}, $k = 0, 1$. Upon reception of the training set D_n, the fusion center numerically computes the vectors of smoothing parameters $\boldsymbol{\lambda}_0^*$ and $\boldsymbol{\lambda}_1^*$ which minimize the leave-one-out estimator for the error probability (33), i.e.

$$(\boldsymbol{\lambda}_0^*, \boldsymbol{\lambda}_1^*) = \underset{(\boldsymbol{\lambda}_0, \boldsymbol{\lambda}_1)}{\operatorname{argmin}} \widehat{P}_e(\boldsymbol{\lambda}_0, \boldsymbol{\lambda}_1). \quad (36)$$

Under this choice of smoothing parameters, the resulting fusion rule is universally consistent, i.e., its probability of error P_e converges with probability one to the probability of error P_e^* of the Bayes optimal fusion rule (9) as the size of the training set D_n tends to infinity [11].

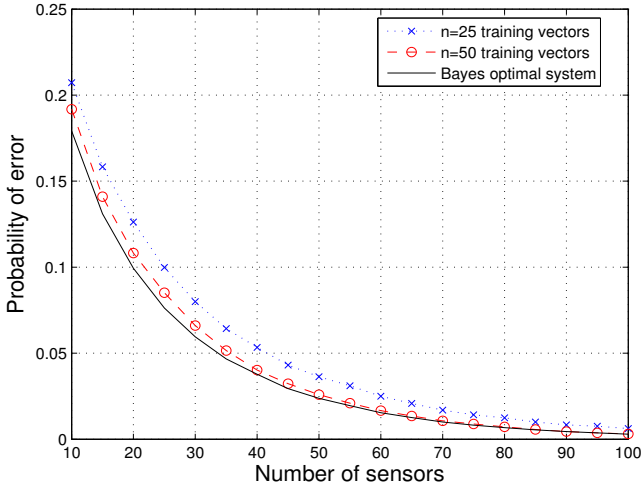

Fig. 3: Independent sensor decisions.

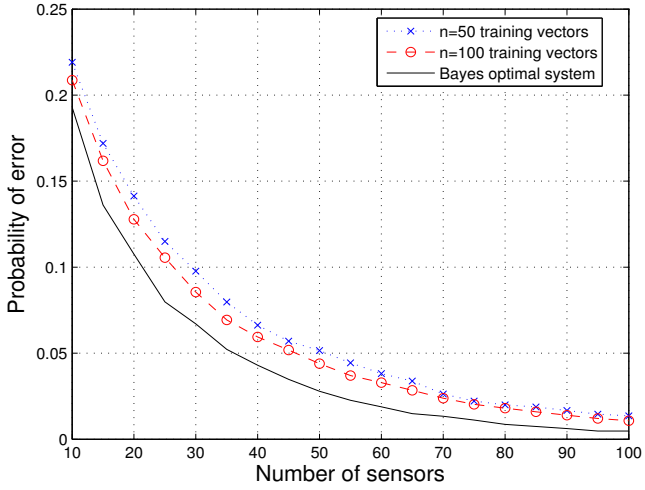

Fig. 4: Correlated sensor decisions with $\beta_j = 0.2$, $j = 1, ..., N$.

5. Numerical Results

In order to obtain numerical results which demonstrate the performance of the calibration algorithms for a wide variety of scenarios, we choose the system parameters in each simulation run to be realizations of *random variables*. In particular, we assume that both the local detection error probabilities P_{f_j} and P_{m_j} as well as the bit-error probabilities ε_j of the BSCs are chosen to be uniformly distributed on the interval $[0, \frac{1}{2}]$, i.e.

$$P_{f_j} \sim \mathrm{U}(0, \tfrac{1}{2}), \quad P_{m_j} \sim \mathrm{U}(0, \tfrac{1}{2}), \quad \varepsilon_j \sim \mathrm{U}(0, \tfrac{1}{2}). \tag{37}$$

We assume that the hypotheses H_0 and H_1 are equally likely to occur, i.e., $\pi_0 = \pi_1 = \frac{1}{2}$. Naturally, the resulting probability of error of the distributed detection system is a random variable because the system is trained on a set of random samples. So we calculate the probability of error by averaging over 1000 independent simulation runs for each combination of sensor network size and number of training vectors.

A. Independent sensor decisions

First, we consider scenarios where the local sensor detection results are independent given the underlying hypothesis. In this case, maximum likelihood parameter estimation as described in Section 3 can be applied resulting in optimal fusion rules when the number of training vectors tends to infinity. However, of particular interest is the detection performance of sensor networks for small to medium size training sets according to real-world conditions. Accordingly, we consider networks of $N = 10, ..., 100$ sensors collecting training sets of $n = 25$ and $n = 50$ labeled samples. The numerical results depicted in Fig. 3 show near optimal performance of our parametric calibration approach even for moderately sized training sets.

B. Correlated sensor decisions

To simulate scenarios in which local sensor detection results are correlated and to evaluate the calibration algorithm from Section 4, efficient methods for generating correlated binary random variables are necessary. We use a method which is based on a model of Oman and Zucker [12]. The multivariate binary distribution of the distorted decision vector $\widetilde{\boldsymbol{U}}$ is modeled by indicator functions of underlying correlated random variables. The thresholds used in the indicator functions correspond to the error probabilities $\widetilde{P}_{f_j}, \widetilde{P}_{m_j}$, $j = 1, ..., N$, and pairwise correlations between local sensor decisions are induced by correlations between the underlying latent variables. Formally speaking, let $Z_0, ..., Z_N$ be i.i.d. continuous random variables which are uniformly distributed on the interval $[0, 1]$, and let $V_1, ..., V_N$ be independent binary random variables with

$$P(V_j = 1) = \beta_j \tag{38}$$

for $j = 1, ..., N$. By initially forming

$$X_j = V_j Z_0 + (1 - V_j) Z_j, \tag{39}$$

we define under hypothesis H_0

$$\widetilde{U}_j = I(X_j \leq \widetilde{P}_{f_j}). \tag{40}$$

and under hypothesis H_1

$$\widetilde{U}_j = I(X_j \leq 1 - \widetilde{P}_{m_j}) \tag{41}$$

for $j = 1, ..., N$, where $I(\cdot)$ is a binary-valued indicator function. With these definitions, it can be shown that for the marginal probabilities of the components of $\widetilde{\boldsymbol{U}}$ holds

$$\begin{aligned} P(\widetilde{U}_j = 1 | H_0) &= \widetilde{P}_{f_j}, \\ P(\widetilde{U}_j = 0 | H_1) &= \widetilde{P}_{m_j}. \end{aligned} \tag{42}$$

Furthermore, we obtain the pairwise correlation coefficients under hypothesis H_0 according to

$$\rho_{jl}^{(0)} = \mathrm{Corr}(\widetilde{U}_j, \widetilde{U}_l | H_0) = \beta_j \beta_l \tau_{jl}^{(0)} \tag{43}$$

where

$$\tau_{jl}^{(0)} = \min \left\{ \sqrt{\frac{\widetilde{P}_{f_j}(1 - \widetilde{P}_{f_l})}{(1 - \widetilde{P}_{f_j})\widetilde{P}_{f_l}}}, \sqrt{\frac{(1 - \widetilde{P}_{f_j})\widetilde{P}_{f_l}}{\widetilde{P}_{f_j}(1 - \widetilde{P}_{f_l})}} \right\}. \tag{44}$$

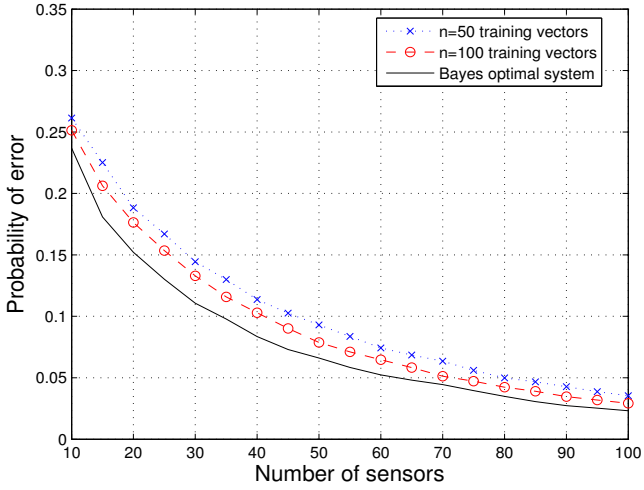

Fig. 5: Correlated sensor decisions with $\beta_j = 0.4$, $j = 1, ..., N$.

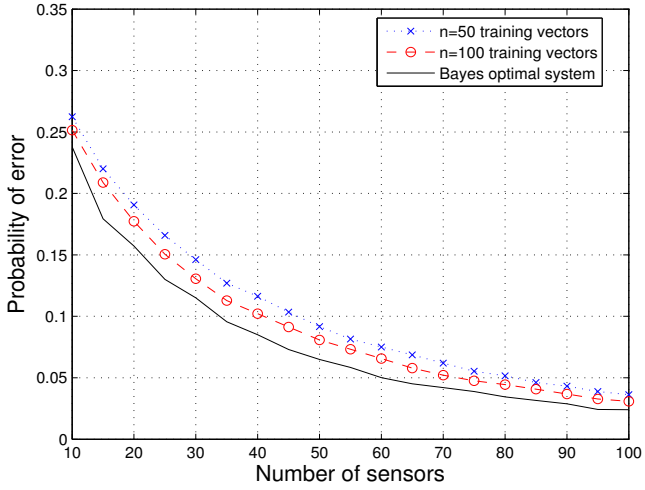

Fig. 6: Correlated sensor decisions with $\beta_j = 0.6$, $j = 1, ..., N$.

Correspondingly, under hypothesis H_1 we obtain the pairwise correlation coefficients

$$\rho_{jl}^{(1)} = \text{Corr}(\widetilde{U}_j, \widetilde{U}_l | H_1) = \beta_j \beta_l \tau_{jl}^{(1)} \tag{45}$$

with

$$\tau_{jl}^{(1)} = \min \left\{ \sqrt{\frac{\widetilde{P}_{m_j}(1-\widetilde{P}_{m_l})}{(1-\widetilde{P}_{m_j})\widetilde{P}_{m_l}}}, \sqrt{\frac{(1-\widetilde{P}_{m_j})\widetilde{P}_{m_l}}{\widetilde{P}_{m_j}(1-\widetilde{P}_{m_l})}} \right\}. \tag{46}$$

By varying the parameters $\beta_1, ..., \beta_N$ from equation (38), the degree of correlation between local sensor decisions can be controlled. The simulation results depicted in Fig. 4 to 6 correspond to values of β_j chosen to be 0.2, 0.4 and 0.6, respectively. If the correlation among sensors is weak ($\beta_j = 0.2$), the results resemble the ones obtained for independent sensors although a higher number of training vectors is necessary because the fusion center now also has to learn the correlation structure of local sensor decisions (Fig. 4). For larger values of β_j there appears to be an overall decrease in performance corresponding to the stronger correlation between sensor decisions. However, it is interesting to note that the gap between the calibrated system and the Bayes optimal one remains approximately of constant size (Fig. 5 and 6).

6. Conclusions

The main concern of the present paper are algorithms for in-situ calibration of sensor networks for distributed detection in the parallel fusion architecture. We distinguished two cases, depending on whether local sensor detection results are correlated or not. Under the assumption of conditionally independent sensor decisions, we employed maximum likelihood parameter estimation in order to obtain fusion rules which achieve minimum detection error for large training sets. In the case of arbitrarily correlated sensor decisions, we used a discrete version of the method of kernel smoothing. Using suitable discrete kernel functions and computing the right smoothing parameters, this procedure also resulted in fusion rules providing optimal detection performance in the asymptotic case. Finally, numerical results obtained by extensive Monte Carlo simulations also show good performance of the in-situ calibration algorithms for moderately sized training sets.

References

[1] D. Li, K. D. Wong, Y. H. Hu, and A. M. Sayeed. Detection, classification, and tracking of targets in distributed sensor networks. In *IEEE Signal Processing Magazine*, 19(2):17–30, March 2002.

[2] R. R. Tenney and N. R. Sandell. Detection with distributed sensors. In *IEEE Trans. Aerosp. Electron. Syst.*, vol. 17, no. 4, pp. 501-510, July 1981.

[3] R. Viswanathan and P. K. Varshney. Distributed detection with multiple sensors: Part I-Fundamentals. In *Proc. IEEE*, vol. 85, no. 1, pp. 54-63, January 1997.

[4] R. S. Blum, S. A. Kassam, and H. V. Poor. Distributed detection with multiple sensors: Part II-Advanced Topics. In *Proc. IEEE*, vol. 85, no. 1, pp. 64-79, January 1997.

[5] S. N. Simic. A learning theory approach to sensor networks. In *IEEE Pervasive Computing*, vol. 2, no. 4, pp. 44-49, 2003.

[6] G. Ferrari and R. Pagliari. Decentralized binary detection with noisy communication links. In *IEEE Trans. Aerosp. Electron. Syst.*, vol. 42, no. 4, pp. 1554-1563, October 2006.

[7] L. Devroye, L. Györfi, and G. Lugosi, *A Probabilistic Theory of Pattern Recognition*, Springer-Verlag, New York, 1996.

[8] E. Drakopoulos and C. C. Lee. Optimum multisensor fusion of correlated local decisions. In *IEEE Trans. Aerosp. Electron. Syst.*, vol. 27, no. 4, pp. 593-606, July 1991.

[9] M. Kam, Q. Zhu, and W. S. Gray. Optimal data fusion of correlated local decisions in multiple sensor detection systems. In *IEEE Trans. Aerosp. Electron. Syst.*, vol. 22, no. 1, pp. 98-101, January 1986.

[10] J. Aitchison and C. G. G. Aitken. Multivariate binary discrimination by the kernel method. In *Biometrika*, vol. 63, no. 3, pp. 413-420, 1976.

[11] G. Tutz. An alternative choice of smoothing for kernel-based density estimates in discrete discriminant analysis. In *Biometrika*, vol. 73, no. 2, pp. 405-411, 1986.

[12] S. D. Oman and D. M. Zucker, Modelling and generating correlated binary variabls. In *Biometrika*, vol. 88, no. 1, pp. 287-290, 2001.