Fault-Tolerant Distributed Classification Based on Non-binary Codes in Wireless Sensor Networks

Tsang-Yi Wang[†], Yunghsiang S. Han[‡], Pramod K. Varshney[*]

***Abstract*— In this letter, we consider the distributed classification problem in wireless sensor networks. The DCSD approach employing the binary code matrix has recently been proposed to cope with the errors caused by both sensor faults and the effect of fading channels. However, the performance of the system employing the binary code matrix could be degraded if the distance between different hypotheses can not be kept large. In this letter, we design the DCSD approach employing the D-ary code matrix when $\log_2 D$ bits local decision information is used, where $D > 2$. Simulation results show that the performance of the DCSD approach employing the D-ary code matrix is better than that of the DCSD approach employing the binary code matrix.**

***Index Terms*— Distributed classification, Wireless sensor networks, Coding, Soft-decision decoding, Decision fusion, Multi-sensor fusion, Fading channel**

I. Introduction

Research on decentralized multiple event or target classifications based on observation in wireless sensor networks has received a great amount of interest recently. In the distributed multiclass classification problem, each local sensor transmits its decision to the fusion center [1], [2] for making the final classification decision. The fusion center can be the cluster head of a cluster-based wireless sensor network. Due to bandwidth and energy limitations in wireless sensor networks, the information bits sent out from each local sensor could be less than $\lceil \log_2 M \rceil$, where M is the number of classes to be distinguished. Also, it has been indicated that, in a wireless sensor network, fault-tolerance capability is critical since sensors can be damaged, blocked, or run out of battery energy [3], [4]. We take the above requirements into consideration in this letter.

Recently, distributed classification fusion approach using error correcting codes (DCFECC) [5] and using soft-decision decoding (DCSD) [6] have been proposed to tolerate several types of faults including stuck-at faults, drained batteries, and channel transmission errors. At the heart of both approaches is the fault-tolerant fusion rule. Unlike the conventional approach that employs the optimal fusion rule [7], the fault-tolerant fusion rule provides enough distance between the decision regions corresponding to different hypotheses by using a code matrix. The observed local decision vectors could still fall into correct decision regions even when several sensor faults are present. The DCSD approach extends the DCFECC approach by using soft decision decoding to combat channel fading.

The original DCSD approach is designed to employ a binary code matrix. However, when the number of sensors is small or the number of hypotheses is large, using binary code matrix may not be enough to keep a large distance between different hypotheses. The classification performance and fault-tolerance capability will be degraded due to the smaller distance between different hypotheses. In this letter, we extend the DCSD approach and employ a D-ary code matrix with $D > 2$ when $\log_2 D$ bit local decision information is used. We call this new approach non-binary DCSD. Note that if D is larger than the number of hypotheses M, then the best code matrix is the repetition code matrix. In this case, we have the largest minimum Hamming distance equal to the number of sensors, N. Thus, in this work we only consider the case that $D < M$.

II. Problem Statement

Let H_ℓ, where $\ell = 0, 1, \ldots, M-1$ and $M \geq 2$, denote the M hypotheses under test at each of the N sensors. Furthermore, the *a priori* probabilities of these M hypotheses are denoted by $P(H_\ell) = P_\ell$, respectively. The observation at each local sensor or detector is represented by y_j, where $j = 1, \ldots, N$. Assume that the distribution function, $P(y_j|H_\ell)$, of y_j under each hypothesis is known. A code matrix $\boldsymbol{C}$ to perform distributed classification fusion is designed by either the simulated annealing or the gradient approach [5] in advance. The code matrix is an $M \times N$ matrix with elements $c_{\ell j}$, $0 \leq \ell \leq M-1$, $1 \leq j \leq N$. $c_{\ell j}$ could be D-ary ($D > 2$) as opposed to be only binary in the problem formulated in our earlier work [5]. Each hypothesis $H_\ell \in \Omega = \{H_0, H_1, \ldots, H_{M-1}\}$ is associated with a row in the code matrix $\boldsymbol{C}$. Each local sensor processes its observations and makes a multilevel D-ary decision $u_j = d$, where $d = 0, \ldots, D-1$, based on the corresponding column of matrix $\boldsymbol{C}$. Since the multilevel local decision rule in the DCSD approach is designed according to either D-ary or binary code matrix $\boldsymbol{C}$, different distance metrics must be employed to measure the distance between the multilevel local decision and the codeword in the given code matrix. These distance metrics will be used to design the optimal local fusion rule but the decoding rule at the fusion center will employ soft-decision decoding.

This work was supported in part by the W.M. Keck Foundation under grant No. CK #1329, by the SUPRIA program of the CASE Center at Syracuse University, and by the NSC of Taiwan, R.O.C., under grants NSC 93-2213-E-305-005.

[†]T.-Y. Wang was with the Dept. of EECS at Syracuse University. Now he is with Graduate Institute of Communication Engineering, National Chi Nan University, Taiwan (E-mail: tswang@ncnu.edu.tw).

[‡]Y. S. Han is with Graduate Institute of Communication Engineering, National Taipei University, Taiwan (E-mail: yshan@mail.ntpu.edu.tw; Phone:+886-2-8674-6887; Fax:+886-2-2674-4448).

[*]P. K. Varshney is with the Dept. of EECS at Syracuse University (E-mail: varshney@syr.edu).

Distance metrics will be defined in Section III. Note that each local sensor makes its decision by itself based on its own observations and is independent of the other sensors. After processing the observations locally, possibly in the presence of faults, the local decisions u_j are mapped to a binary signal vector $\boldsymbol{b}_j = (b_{j1}, \ldots, b_{jS})$, where $S = \lceil \log_2 D \rceil$ is the number of bits to represent the local decision $u_j = d, d = 0, \ldots, D-1$. In this letter, we assume that all local decisions, u_j, take values from 0 up to $D-1$. For instance, a four-level local decision is transmitted by means of one of the 2-bit binary signal vectors, $\{11, 10, 01, 00\}$. These binary signal vectors are transmitted to the fusion center over parallel channels that are assumed to undergo independent fading. We further make the assumption of phase coherent reception. Hence, the effect of fading channels is further simplified as a real scalar multiplication given the transmitted signal.

We assume that binary antipodal signalling is employed for transmission and results in a received vector at the fusion center consisting of real numbers, $\boldsymbol{R} = (\boldsymbol{r}_1, \boldsymbol{r}_2, \ldots, \boldsymbol{r}_N), \boldsymbol{r}_j = (r_{j1}, \ldots, r_{jS})$, where $j = 1, \ldots, N$. The r_{js}, for $s = 1, \ldots, S$, can be expressed as $r_{js} = \alpha_{js}(-1)^{b_{js}}\sqrt{E_b} + n_{js}$, where E_b is the energy per channel bit and n_{js} is a noise sample from a Gaussian process with two-sided power spectral density $N_0/2$. α_{js} is the attenuation factor determined by the fading type.

III. The Non-binary DCSD scheme

For simplicity, the derivation of local decision rules is based on receiving multilevel quantization (hard) decisions instead of the real numbers at the fusion center. The real vector is used, however, during the design of the non-binary DCSD fusion rule (decoding rule).

A. Optimal local decision rules

Let the Hamming distance measure between $\boldsymbol{u} = (u_1, \ldots, u_N)$ and $\boldsymbol{c}_\ell = (c_{\ell 1}, \ldots, c_{\ell N}), \ell = 0, \ldots, M-1$, be defined as $d_D(\boldsymbol{u}, \boldsymbol{c}_\ell) = \sum_{j=1}^{N} d_j^D(u_j, c_{\ell j})$, where $d_j^D(u_j, c_{\ell j}) = 0$ if $u_j = c_{\ell j}$; otherwise $d_j^D(u_j, c_{\ell j}) = 1$. Define $L^{\ell}_{i_1, i_2, \ldots, i_N}$, where $i_1, i_2, \ldots, i_N \in \{0, 1, \ldots, D-1\}$, as the cost that the received word at the fusion center $\boldsymbol{u}$[1] equals $(i_1, i_2, \ldots, i_N)$ and the true hypothesis is H_ℓ. These costs $L^{\ell}_{i_1, i_2, \ldots, i_N}$ can be determined by the decision regions of codewords. According to the designed code matrix, the decision region Z of a codeword $\boldsymbol{c} \in \boldsymbol{C}_w$ is given as follows:

$$Z(\boldsymbol{c}) = \{\boldsymbol{u} | d_D(\boldsymbol{u}, \boldsymbol{c}) \le d_D(\boldsymbol{u}, \boldsymbol{c}') \text{ for all } \boldsymbol{c}' \in \boldsymbol{C}_w\},$$

where $\boldsymbol{C}_w = \{\boldsymbol{c}_\ell | \ell = 0, \ldots, M-1\}$ is the set of all codewords, i.e., all rows of the code matrix. In order to minimize the probability of misclassification, set $L^{\ell}_{i_1, \ldots, i_N} = 0$ if $(i_1, \ldots, i_N)$ is in the decision region of $\boldsymbol{c}_\ell$ that is the row of $\boldsymbol{C}$ corresponding to the hypothesis H_ℓ; otherwise set $L^{\ell}_{i_1, \ldots, i_N} = 1$. Whenever a received vector $(i_1, \ldots, i_N)$ simultaneously belongs to decision regions of $c_{k_0}, c_{k_1}, \ldots, c_{k_{q-1}}$, where $q > 1$, for all $\ell = 0, \ldots, q-1$, set $L^{\ell}_{i_1, \ldots, i_N} = (1-1/q)$, i.e., we assume the fusion center randomly picks one codeword among the codewords which are at the same distance from the received word $\boldsymbol{u}$.

According to the costs assigned above, the probability of error can be minimized if we set the local decision rule at sensor k as

$$P(u_k = i_k | y_k) = \begin{cases} 1, & \text{if } \boldsymbol{I}_k^*(i_k) \le \boldsymbol{I}_k^*(m) \text{ for all } i_k \\ & \text{and } m \text{ such that } i_k \ne m, \text{ and} \\ & i_k, m = 0, \ldots, D-1; \\ 0, & \text{otherwise,} \end{cases} \tag{1}$$

where,

$$\begin{aligned} \boldsymbol{I}_k^*(i_k) \;=\; & \sum_{\ell} P(y_k | H_\ell) \sum_{i_1, \ldots, i_{k-1}, i_{k+1}, \ldots, i_N} P_\ell \times \\ & P(u_1 = i_1 | H_\ell) \times \cdots \times P(u_{k-1} = i_{k-1} | H_\ell) \times \\ & P(u_{k+1} = i_{k+1} | H_\ell) \times \cdots \times P(u_N = i_N | H_\ell) \times \\ & L^{\ell}_{i_1, \ldots, i_k, \ldots, i_N}, \end{aligned} \tag{2}$$

under the assumption of conditionally independent observations given their hypotheses.

B. Non-binary DCSD fusion (decoding) rule

As mentioned earlier, the essence of the fusion process in the DCFECC approach is decoding. This coding structure enables us to consider the received vector at the fusion center as a codeword transmitted collectively from all local sensors. The non-binary DCSD fusion rule is then able to jointly consider the local decision rules and word-by-word decoding to achieve robust system performance by providing sensor fault-tolerance capability and channel error correction.

By decoding the received vector using the MAP criterion and assuming equal prior probability of each hypothesis, we have the following fusion rule: Given the received vector $\boldsymbol{R}$, set $\hat{\boldsymbol{c}} = \boldsymbol{c}_\ell \in \boldsymbol{C}_w$, if

$$P(\boldsymbol{R} | \boldsymbol{c}_\ell) \ge P(\boldsymbol{R} | \boldsymbol{c}_k), \text{ for all } \boldsymbol{c}_k \in \boldsymbol{C}_w. \tag{3}$$

Assuming conditional independence of observations at the sensors and discrete memoryless channels between local decision outputs and the fusion center, (3) can be formulated as

$$\prod_{j=1}^{N} P(\boldsymbol{r}_j | c_{\ell j}) \ge \prod_{j=1}^{N} P(\boldsymbol{r}_j | c_{kj}), \text{ for all } \boldsymbol{c}_k \in \boldsymbol{C}_w.$$

The above equation can be rewritten as

$$\sum_{j=1}^{N} \ln \frac{\sum_{d=0}^{D-1} P(\boldsymbol{r}_j | u_{jd}) P(u_{jd} | c_{\ell j})}{\sum_{d=0}^{D-1} P(\boldsymbol{r}_j | u_{jd}) P(u_{jd} | c_{kj})} \ge 0, \text{ for all } \boldsymbol{c}_k \in \boldsymbol{C}_w. \tag{4}$$

since the received $\boldsymbol{r}_j$ only depends on the local decision u_{jd} and does not depend on the code matrix we designed.

IV. Performance evaluation

In this section, the performance of the non-binary DCSD using D-ary code matrices, where $D = 4$, is evaluated and then compared with that of the DCSD employing binary code matrices. Since $D = 4$ we only consider the case of 2-bit information in the simulation.

[1]The received word $\boldsymbol{u}$ is assumed to be a hard-decision result. That is, $u_i \in \{0, 1, \ldots, D-1\}$.

A system with a fusion center and seven independent local sensors are considered to identify seven equally likely hypotheses H_0,H_1,H_2,H_3,H_4,H_5, and H_6.The probability density function for each hypothesis is assumed to be a Gaussian distribution with the same variance ($\sigma^2 = 1$) but with different means $-3V$,$-2V$,$-V$,0,V,$2V$, and $3V$, respectively. We assume that all the sensor measurements are identically distributed. The observation signal-to-noise ratio (OSNR) at each local sensor is defined as $20\log_{10} V$. Also, the total energy E output from the local sensor nodes is fixed, and $E = S \times E_b$, where E_b is the energy per channel bit. The decisions of local sensors are transmitted over Rayleigh fading channels to the fusion center. The channel signal-to-noise ratio (CSNR) is defined as $\gamma = E_b/N_0 \times E[\alpha_{js}^2]$. In this evaluation, the Gauss-Seidel algorithm [1] is used to compute the local decision rules in (1).

The D-ary code matrix and the binary code matrix employed in this evaluation are designed by simulated annealing and are given in Code Matrices I and II, respectively:

$$\text{Code Matrix I: } (11556, 7693, 7276, 2008, 1923, 7209, 9487)$$

$$\text{Code Matrix II: } (25, 112, 99, 57, 102, 51, 51).$$

The referred code matrices are represented as a vector of M bit integers. Each integer z_j corresponding to column j in the code matrix represents a column vector in the code matrix, and can be expressed as $z_j = \sum_{\ell=0}^{M-1} t_{\ell j} \times Q^\ell$, where $Q = D$ if D-ary code matrix is used and $Q = 2$ if binary code matrix is employed. It is easy to see that the minimum Hamming distance between any two codewords in the D-ary code matrix is 4. However, the minimum Hamming distance between any two codewords in the binary code matrix is only 2. The performance of both schemes are evaluated in both fault-free and faulty situations. We consider the stuck-at fault in this evaluation. When the stuck-at fault occurred at a particular sensor node, we assume that the sensor always makes the decision 3, i.e. $u_1 = 3$ and sends the 2-bit binary signal vector 11 to the fusion center.

Figs. 1 and 2 show the performance of the system that employ the D-ary code matrix and binary code matrix in sensor-fault free case, one stuck-at fault (sensor 1) case, and two stuck-at faults (sensors 1 and 2) case. Fig. 1 shows that the performance comparison when the transmission channel has $E/N_0 = 5$ dB. Fig. 2 shows the performance comparison when the OSNR is fixed at 5 dB. From the result of both figures, one can see that the performance of the system employing a D-ary code matrix is better than that of the system employing the binary code matrix in both the sensor-fault free situation and sensor faults case. The performance difference becomes more and more apparent as the number of sensor faults increases.

V. Conclusions

We extended the DCSD approach by using a D-ary code matrix, where $D > 2$. By employing the D-ary code matrix, the distance between different codewords corresponding to their hypotheses can be larger as compared with using the binary code matrix. Taking advantage of this, the classification performance and fault-tolerance capability can be improved even further. It is shown through computer simulations that the performance of the non-binary DCSD approach employing the D-ary code matrix is better than that of DCSD employing the binary code matrix.

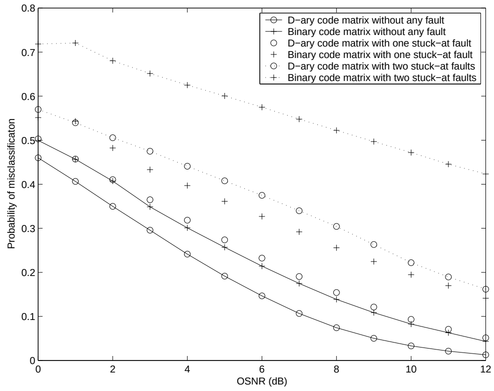

Fig. 1. Performance comparison of the non-binary DCSD with D-ary code matrix and the DCSD with binary code matrix, when stuck-at faults are considered. E/N_0 is at 5 dB.

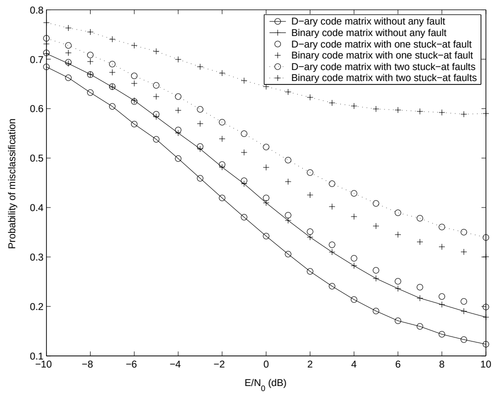

Fig. 2. Performance comparison of the non-binary DCSD with D-ary code matrix and the DCSD with binary code matrix when stuck-at faults are considered. OSNR is at 5 dB.

References

[1] P. K. Varshney, *Distributed Detection and Data Fusion.* New York: Springer, 1997.

[2] F. A. Sadjadi, "Hypotheses testing in a distributed environment," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 22, no. 2, pp. 134–137, March 1986.

[3] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "A survey on sensor networks," *IEEE Communications Magazine*, pp. 102–114, August 2002.

[4] L. Dan, K. D. Wong, H. H. Yu, and A. M. Sayeed, "Detection, classification, and tracking of targets," *IEEE Signal Processing Magazine*, vol. 19, pp. 17–29, March 2002.

[5] T.-Y. Wang, Y. S. Han, P. K. Varshney, and P.-N. Chen, "Distributed fault-tolerant classification in wireless sensor networks," *IEEE Journal of Selected Areas in Communications*, vol. 23, no. 4, April 2005.

[6] T.-Y. Wang, Y. S. Han, and P. K. Varshney, "A combined decision fusion and channel coding scheme for fault-tolerant classification in wireless sensor networks," in *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing 2004 (ICASSP '04)*, vol. 2, Montreal, CA, May 2004, pp. 1073 – 1076.

[7] Z. Chair and P. K. Varshney, "Optimal data fusion in multiple sensor detection systems," *IEEE Trans. Aerosp. Electron. Syst.*, vol. AES-22, pp. 98–101, January 1986.