Decentralized Detection In Wireless Sensor Networks

Milad Kharratzadeh

Department of Electrical & Computer Engineering
McGill University
Montreal, Canada

April 2011

Statistical Detection and Estimation
Term Project

Abstract

Wireless sensor networks consist of small nodes which are capable of sensing, computation, and communication. The initial goal of a sensing system is to detect the events of interest. In this project, we study decentralized detection in sensor networks. First, we will review the classical decentralized detection framework, in which a set of spatially distributed sensor nodes receives information about the state of nature. Because of the channel bandwidth constraint, each node sends a quantized function of its observations to a central node, called fusion center. The fusion center then decides on one of the alternative hypotheses. Designing the local functions of individual sensor nodes and also the decision rule at the fusion to minimize the probability of error is of interest in this context [1, 2, 3].

The classical framework does not adequately take into account important features of sensor network architecture and of the wireless channel. Next, we will review some works that consider these features. The type of channel imposes some restrictions on the messages from sensors to fusion. Two different types of wireless channel between the sensors and the fusion center is studied here: multiple access channel [4] and broadcast channel [5]. In multiple access channel the total amount of data sent by all sensors is bounded by the channel capacity [4], and in broadcast channel, there is no central node and all nodes try to reach a consensus by successive retesting and rebroadcasting of the updated decision [5]. A common architecture for wireless sensor networks is the tree configuration. Here, we will review the optimal strategies for sensor nodes (i.e. local functions, which are based on the messages received from predecessors and own observations) and also the decision rule at fusion in terms of optimizing the error exponent [6].

In the end, as the novel part of this project we propose a totally decentralized detection method which is based on gossiping. In this method, we assume that nodes are only capable of short-range communications. In each round of the algorithm. sensors communicate two-by-two (locally) and update their likelihood ratio functions. We will prove that the proposed algorithm converges and its solution is exactly the same as the centralized scheme.

Contents

Chapter 1

Introduction

Wireless Sensor Networks consist of a set of sensors, which are capable of sensing, computation, and communication and are spatially distributed in order to cooperatively monitor physical or environmental conditions. The ability to detect events of interest is essential to the success of emerging sensor network technology. Detection often serves as the initial goal of a sensing system. For example in applications where we want to estimate attributes such as position and velocity, the first step is to ascertain the presence of an object. Moreover, in some applications such as surveillance, the detection of an intruder is the only purpose of the sensor system. In situations where the peripheral nodes do some preprocessing on their observation before sending data to a central node (called fusion center), the corresponding decision making problem is termed decentralized detection.

Assume that there are M hypotheses on the state of the environment and each one of the sensors observe some relevant information about it. In a centralized scheme, each sensor transmits all of its observation (without any additional processing) to the fusion center, which solves a classical hypotheses testing problem and decides on one of the M hypotheses. However, in a classical decentralized scheme, each sensor will do some preprocessing and send a summary of its observation, which is chosen from a finite alphabet, to the fusion center. Then, the fusion center will decide on one of the M hypotheses, based on the messages it has received.

The described centralized and decentralized schemes, differ in some aspects. First, it is clear that the performance of the decentralized scheme is suboptimal in comparison with the centralized scheme due to loss of information in the nodes local preprocesses. On the other hand the communication requirements of decentralized scheme is much smaller than those of centralized one. The reason is that instead of sending raw voluminous observation, each node sends a summary of its observation, taken from a finite alphabet. So, in brief, the decentralized detection offers a great reduction in communication requirements, at the expense of some performance reduction. However, it turns out that the performance reduction is often negligible [2].

In decentralized scheme, while the fusion center faces a classical hypotheses testing problem (if we look at the messages received from other nodes as its observations), the problem is more complex for the peripheral sensors. One may expect that each sensor should decide independently and make decision only based on its own observation and use its own likelihood ratio test. This is not true in general. When the detectors decide in a way to achieve a system-wide optimization, they often use different strategies than in cases where the joint costs of their decisions separates into a cost for each. Even under a conditional independence assumption (which means that the observations of different sensors are independent from each other under the same hypothesis), finding optimal decision-making algorithms (based on the observation) at the sensor nodes remains, in most cases, a difficult task. This optimization problem is known to be tractable only under restrictive assumptions regarding the observation space and the topology of the underlying network [2].

In chapter 2 we will provide a detailed study of the classical decentralized detection problem. First, we will review the very basic decentralized detection problem for the binary hypotheses testing [1], in order to get an insight about the decentralized scheme. Although this is a very special case, it will help us developing the more general cases, more easily. Then, we will review the general problem of the decentralized detection under Bayesian criterion [2]. A precise formulation of the problem is provided and also the optimal solutions for the case of conditionally independent observations are characterized. Finally, we will consider the Neyman-Pearson variant of the problem [3]. Again we will formulate the problem and characterize the optimal strategy. It will be seen that unlike the centralized detection, where the Bayesian and Neyman-Pearson problems are almost equivalent, the situation is more complex in decentralized detection.

As we will see, the classical framework does not adequately take into account important features of sensor network architecture and of the wireless channel. In chapter 3, we will study two special type of wireless channels between the nodes and the fusion center. First, we will consider a multiple access channel which imposes a constraint on the total capacity of the channel [4]. Then, we consider a broadcast channel, where the nodes try to reach a consensus without having a central node (Parley algorithm) [5]. Another feature that was not considered in classical framework is the architecture of the network. Here, we will briefly review the tree architecture for the network [6].

Finally, in chapter 4, which is the novel part of this project, we will propose a gossip-based method for decentralized detection. As Parley algorithm, there is no central node (fusion center) here, but the huge difference is that in our algorithm, the nodes communicate locally rather than through a broadcast channel. In our method, the nodes try to update their likelihood ratios and also revise their decision in each round of gossiping. We will show that the proposed method converges and the solution matches the solution of the centralized scheme. To the best of our knowledge, this is a novel method.

Chapter 2

Classical Framework of Decentralized Detection

2.1 Decentralized Binary Hypotheses Testing

In this section we review the seminal work of Tenney and Sandell [1], which is the first work to consider a decentralized detection system in which, each of the peripheral sensors send a summary of its observation (using a transmission function) to a fusion center. They studied the decentralized detection for the case of binary hypotheses. Although this is a very specific and basic variant of the problem, it provides valuable insights about it that will help us in exploring the more general cases in the following sections.

For the structure of Fig. 2.1, the problem of decentralized binary hypothesis testing is stated as follows. The two hypotheses are H_1 and H_2 with prior probabilities $\Pr(H_1)$ and $\Pr(H_2)$, respectively. Each of these hypotheses induces a different joint probability distribution for the observations:

$$\text{Under } H_1 \quad : \quad \Pr(y_1|H_1) \tag{2.1}$$

$$\text{Under } H_2 \quad : \quad \Pr(y_1|H_2) \tag{2.2}$$

where Y_1 and Y_2 are the observations of each of the two peripheral nodes, and belong to the observation space (the set of all possible observations). As it is clear from the Fig. 2.1, we will consider only the case of two peripheral sensors and one fusion center, which does not have any observation itself here. Also we assume that the activity of fusion center is fixed a priori, which means that the decision of the fusion center is a deterministic function of the messages it receives from peripheral sensors.

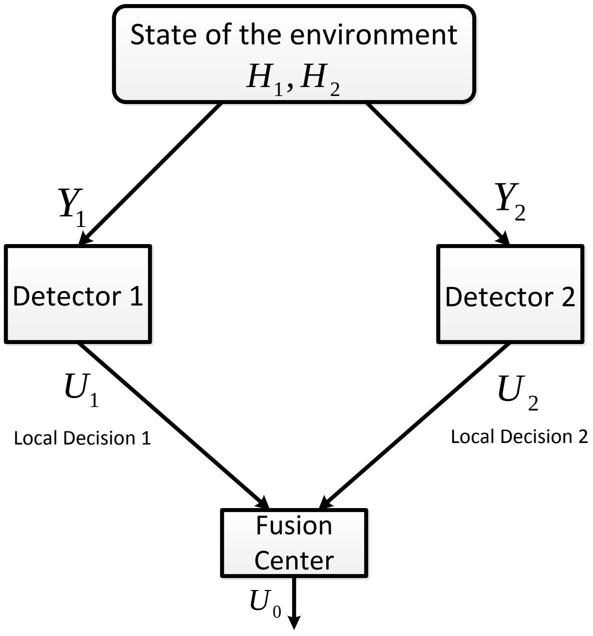

Fig. 2.1 Decentralized binary hypotheses testing

Each of the detectors, based on its observation, will choose one message, U_i, from the set $\{1, 2\}$ and send it to the fusion center. In other words:

$$u_i = \begin{cases} 1, & H_1 \text{ have been detected} \\ 2, & H_2 \text{ have been detected} \end{cases}$$

where u_i, for $i = 1, 2$ is the transmitted message from detector i to fusion center. A randomized decision rule can be defined by the conditional probability distribution function $\Pr(u_i = 1|y_i)$ and $\Pr(u_i = 2|y_i) = 1 - \Pr(u_i = 1|y_i)$ which depend only the observation y_i for $i = 1, 2$. Here, we consider Bayesian criterion, and thus the objective is to minimize the expectation of a cost function. Note that the final decision is made by the fusion center and the variable that interprets the final decision of the total system is u_0 (i.e. we should define our cost as a function of u_0). However, as mentioned before, in this example we assume the decision rule at fusion center is fixed a perior and thus u_0 is a deterministic function of u_1 and u_2. Therefore, we can define the cost as a function of only u_1 and u_2:

$$C(u_1, u_2, H_i) : \{1, 2\} \times \{1, 2\} \times \{H_1, H_2\} \rightarrow \mathbb{R} \tag{2.3}$$

with $C(u_1, u_2, H_i)$ being the cost incurred for detector 1 choosing u_1, and detector 2 choosing u_2, while H_i is true. As told before, the objective is to find a decision strategy (in other words finding $\Pr(u_i = 1|y_i)$ and $\Pr(u_i = 2|y_i)$ for all $y_i, i = 1, 2$ in observation space) such that it minimizes the expectation of cost function, i.e. $E[C(U_1, U_2, H)]$.

We know that the optimal Bayesian decision rule for the *centralized* binary hypotheses testing problem is a deterministic likelihood ratio test with a threshold based on the prior probabilities and the cost function. We can derive the solution for the decentralized case, in a somewhat similar manner. To do so, we focus on one detector at a time and try to adjust its decision rule to minimize the expectation of cost function. This method is called person-by-person minimization. If a strategy is optimal it must also be person-by-person optimal (i.e. it is a necessary but not sufficient condition). So, let's begin by expanding $E[C(U_1, U_2, H)]$:

$$\begin{aligned} E[C(U_1, U_2, H)] &= \sum_{H,u_1,u_2} \int_{y_1,y_2} \Pr(u_1, u_2, y_1, y_2, H) \cdot C(u_1, u_2, H) \\ &= \sum_{H,u_1,u_2} \int_{y_1,y_2} \Pr(H)\Pr(y_1, y_2|H)\Pr(u_1|y_1)\Pr(u_2|y_2) \cdot C(u_1, u_2, H) \end{aligned}$$

In the second step, we used the total probability theorem and also the fact that u_1 only depends on y_1 and u_2 only on y_2, and they are independent from each other (i.e. $\Pr(u_1, u_2|y_1, y_2, H) = \Pr(u_1|y_1, y_2, H) + \Pr(u_2|y_1, y_2, H) = \Pr(u_1|y_1) + \Pr(u_2|y_2)$). Now, if we sum over u_1, we will have:

$$\begin{aligned} \sum_{H,u_2} \int_{y_1,y_2} \Pr(H)\Pr(u_2|y_2)\Pr(y_1, y_2|H) \cdot [&\Pr(u_1 = 1|y_1)C(1, u_2, H) \\ &+(1 - \Pr(u_1 = 1|y_1))C(2, u_2, H)] \end{aligned}$$

As we said before, we focus only on one detector at a time. If we consider detector 1, the terms that do not contain the variable u_1, are characterized as constants and can be eliminated from the objective function (which is to be minimized). Thus, we have the following equivalent function to minimize:

$$\int_{y_1,y_2} \Pr(u_1 = 1|y_1) \sum_{H,u_2} \Pr(H)\Pr(u_2|y_2)\Pr(y_1, y_2|H) \cdot [C(1, u_2, H) - C(2, u_2, H)] \quad (2.4)$$

This can be minimized by the following decision rule:

$$\Pr(u_1 = 1|y_1) = \begin{cases} 0, & \text{if } \sum_{H,u_2} \Pr(H)\Pr(u_2|y_2)\Pr(y_1, y_2|H) \cdot [C(1, u_2, H) - C(2, u_2, H)] \geqslant 0 \\ 1, & \text{else} \end{cases}$$

So, it turns out that we have a deterministic decision rule which is based on the sum. In brief, the final decision rule of detector 1 is the following threshold test:

$$\sum_{H,u_2} \Pr(H)\Pr(u_2|y_2)\Pr(y_1, y_2|H) \cdot [C(1, u_2, H) - C(2, u_2, H)] \underset{H_1}{\overset{H_2}{\gtrless}} 0$$

If we expand the sum over H and assume $C(2, u_2, H_1) > C(1, u_2, H_1)$, which means that the error is more costly than the correct detection, then we have:

$$\frac{\Pr(H_1)\Pr(y_1|H_1)}{\Pr(H_2)\Pr(y_2|H_2)} \underset{H_1}{\overset{H_2}{\gtrless}} \frac{\sum_{u_2} \int_{y_2} \Pr(y_2|y_1, H_2)\Pr(u_2|y_2)[C(1, u_2, H_2) - C(1, u_2, H_2)]}{\sum_{u_2} \int_{y_2} \Pr(y_2|y_1, H_1)\Pr(u_2|y_2)[C(1, u_2, H_1) - C(1, u_2, H_1)]} \qquad (2.5)$$

The left hand side is the likelihood ratio for y_1. However, the right hand side is not a simple threshold constant and depends on $\Pr(u_2|y_2)$ which is the decision rule of the detector 2 and $\Pr(y_2|y_1, H_i)$ which is the conditional density of detector 2 observation. This causes a kind of difficulty that we have not seen before in the centralized scheme: finding the optimal strategy requires solving (2.5) and its companion form (for detector 2) through the coupled equations. Therefore, the optimal solution is NOT a likelihood ratio test in general. However, matters simplify if we assume that the observations are conditionally independent, which means that the observations are statistically independent of each other under the same hypothesis:

$$\begin{aligned}\Pr(y_2|y_1, H) &= \Pr(y_2|H) \\ \Pr(y_1|y_2, H) &= \Pr(y_1|H)\end{aligned}$$

This assumption is satisfied in problems of detecting a known signal, when each sensor's observation is corrupted by independent noise. However, it is violated in problems of detecting an unknown signal or when the noises are dependent. So, if the above assumption is satisfied, the dependence of the right hand side of (2.5) on y_1 is removed and the threshold to which we compare the likelihood ratio becomes only a function of the decision rule of detector 2 ($\Pr(u_2|y_2)$). Therefore, under the conditional independence assumption, **the optimal strategy for the decentralized binary hypotheses testing problem has each detector implementing a likelihood ratio test, using a threshold which is derived from the decision rule of the other detector.**

If we call the thresholds for detectors 1 and 2, t_1 and t_2 respectively, we have:

$$\begin{aligned}t_1 &= \frac{\int_{y_2} \Pr(y_2|H_2)[C(1,2,H_2) - C(2,2,H_2)] + \Pr(u_2 = 1|y_2) \cdot C_1}{\int_{y_2} \Pr(y_2|H_1)[C(2,2,H_1) - C(1,2,H_1)] + \Pr(u_2 = 1|y_2) \cdot C_2} \\ &\triangleq f_1(t_2)\end{aligned}$$

where

$$C_1 = C(1,1,H_2) - C(2,1,H_2) - C(1,2,H_2) + C(2,2,H_2) \tag{2.6}$$

$$C_2 = C(2,1,H_1) - C(1,1,H_1) - C(2,2,H_1) + C(1,2,H_1) \tag{2.7}$$

And similarly $t_2 = f_2(t_1)$. Basically, what these two equations say is that we should solve the simultaneous equations to find the thresholds t_1 and t_2 in the above setting.

As we said before the problem that we considered in this chapter was a very specific example of the decentralize detection which was studied by Tenney and Sandell [1]. However, as we will see in the next sections most of the achieved results are valid for the general decentralized detection problem. In the following two sections we will study Bayesian and Neyman-Pearson formulation of the decentralized detection problem.

2.2 Bayesian Framework

In this section we study the Bayesian formulation of the decentralized detection problem. The objective, as the centralized scheme, is the minimization of the expectation of a cost function. In the following, first, we will propose a formulation of the problem. Then, we will characterize the optimal strategy for the case of conditionally independent sensor's observation. Here, we will review a part of the work of John Tsitsiklis [2]. Also, we will use some of the contents from [1].

2.2.1 The Main Model and Problem Formulation

In this section and the next one (Neyman-Pearson Framework), we assume that all sensors send their messages ("summary" of their observations) to the fusion center directly (i.e. Star topology). We consider the decentralized organization with the following basic parameters:

- M: The number of hypotheses ($M \geqslant 2$)
- N: The number of peripheral sensors ($N \geqslant 1$)
- D: The number of distinct messages that each sensor can send to the fusion center, the size of the alphabet that each sensor chooses its message from ($D \geqslant 2$)

So, we have M hypotheses $H_1, \ldots, H_M$, with prior probabilities $\Pr(H_1), \ldots, \Pr(H_M)$, respectively. There are $N+1$ sensors $S_0, S_1, \ldots, S_N$. Each sensor S_i receives an observation Y_i which is a random variable taking values in a set $\mathcal{Y}_i$ (which sometimes is called as observation space). We call sensor S_0 the fusion center. It plays an important role in decentralized detection problem. All the other sensors, $S_1, \ldots, S_N$ (which are also called peripheral sensors), upon receiving a realization y_i of the random variable Y_i, evaluates a message $u_i = \gamma_i(y_i) \in \{1, \ldots, D\}$, and transmits it to the fusion center, S_0. We call the function $\gamma_i : \mathcal{Y}_i \rightarrow \{1, \ldots, D\}$, as the decision rule of the sensor i. The fusion center receives all these messages as well as its own observation (y_0), and using its own decision rule (which is also called fusion rule), $\gamma_0 : \mathcal{Y}_0 \times \{1, \ldots, D\}^N \rightarrow \{1, \ldots, M\}$, decide on one of the M possible hypotheses.

A collection $\gamma = (\gamma_0, \gamma_1, \ldots, \gamma_N)$ of decision rules are referred to as a strategy. The set of all strategies is denoted as Γ. As the centralized case, we also can have randomized decision rules. In the case where all the nodes use randomization, there are two variants:

- **Independent randomization:** Each sensor i has a finite set of possible decision rules $\{\gamma_i^{(k)} | k = 1, \ldots, K_i\}$ and uses decision rule $\gamma_i^{(k)}$ with probability $p_{i,k}$ (we must have $\sum_{k=1}^{K_i} p_{i,k} = 1$). Moreover, the selection of the decision rule for each sensor is independent of all other decision rules for other sensors. Let $\overline{\Gamma}$ be the set of all independently randomized strategies.

- **Dependent randomization:** In this case, there is a finite set of deterministic strategies $\{\gamma^{(k)} | k = 1, \ldots, K\}$, and strategy $\gamma^{(k)} = (\gamma_0^{(k)}, \gamma_1^{(k)}, \ldots, \gamma_N^{(k)})$ is used with probability p_k (we must have $\sum_{k=1}^{K} p_k = 1$). Thus, in this case the selections of decision rules by the different sensors are dependent. Let Γ^* be the set of all dependently randomized strategies.

From the above definitions we have $\Gamma \subset \overline{\Gamma} \subset \Gamma^*$. The reason is that if we put for some specific k, $p_{i,k} = 1$ and $p_{i,j} = 0, j \neq k$, and do the same thing for other sensors, then we will get a simple strategy. So the simple strategy is a particular case of independently randomized strategy and we have $\Gamma \subset \overline{\Gamma}$. Next, if we put $p_k = \prod_{i=0}^{N} p_{i,k_i}$ in a dependently randomized strategy, we will get an independently randomized strategy. Thus, $\overline{\Gamma} \subset \Gamma^*$.

When we fix a decision rule γ_i for sensor S_i, the message u_i which is transmitted to the fusion center from this sensor is a realization of a random variable U_i defined by $U_i = \gamma_i(Y_i)$. Obviously, the distribution of U_i depends on the decision rule and the distribution of the observation, Y_i. Once a strategy is fixed, the decision of the fusion center becomes a random variable U_0 defined by $U_0 = \gamma_0(Y_0, U_1, U_2, \ldots, U_N)$. The same thing is true for the randomized decision rules.

In the Bayesian formulation, we are given a cost function

$$C(U_0, U_1, \ldots, U_N, H) : \{1, \ldots . M\} \times \{1, \ldots, D\}^N \times \{H_1, \ldots, H_M\} \to \mathbb{R} \tag{2.8}$$

where $C(u_0, u_1, \ldots, u_N, H_i)$ is the cost associated with the event that hypothesis H_i is true, the messages from the peripheral sensors are $u_1, \ldots, u_N$, and the fusion center decision is u_0. Our objective in Bayesian framework is to find a strategy, γ that minimizes the expectation of the cost function: $J(\gamma) \triangleq E[C(U_0, U_1, \ldots, U_N, H)]$. Thus:

$$J(\gamma) = \sum_{i=1}^{M} \Pr(H_i) \cdot E[C(\gamma_0(Y_0, \gamma_1(Y_1), \ldots, \gamma_N(Y_N)), \gamma_1(Y_1), \ldots, \gamma_N(Y_N), H_i)|H_i]$$

As in the centralized case we do not consider the randomized strategies for Bayesian framework. The reason, briefly, is that the minimum of the linear combinations of some variables cannot be less than the minimum of the variables. So, in the following we only consider deterministic strategies.

Remark 1. Note that the cost function defined above is a function of the fusion decision as well as the peripheral nodes messages. However, for different purposes we can define different cost functions that only depend on some of decisions. For example, if C only depends on U_0, the performance of the system is judged on the basis of the fusion center's decision. The minimum probability of error criterion for the final decision lies in this type of cost functions. As another case, when $D = M$, we may wish to interpret each sensor's message as a local decision, based on the true hypothesis. Then, with the suitable choice of C, we can penalize incorrect decisions by the fusion center as well as the peripheral sensors. As an extreme case, the cost function might be independent of the fusion center's decision and be only a function of the decisions of peripheral nodes and we only need to optimize with respect to $\gamma_i, i = 1, \ldots, N$. This can happen in the case of a priori fixed fusion decision rule (as we saw in section 2.1).

Remark 2. When $M = D$, it may seem right that the decision of the peripheral nodes should interpret the true hypothesis. Although we can enforce this by choosing an appropriate cost function (as discussed in previous remark), this is not true in general. For instance, when we want to minimize the fusion's probability of error, the optimal sensor messages are very poor when viewed as local decisions.

2.2.2 Finding The Optimum Solution

As we saw in section 2.1, if we assume that the observations are conditionally independent, then we can find the optimal solution by doing likelihood ratio tests. Also in this section, we assume that this condition holds. Moreover, as in section 2.1, we perform a person-by-person optimization on the peripheral sensors to find the optimal solutions. This means that at each step, we fix some $i \neq 0$ and suppose that γ_j has been fixed for all $j \neq i$. Then we try to find the γ_i that minimizes the expectation of cost function. As we said in section 2.1, if a strategy is optimal, then it must also be person-by-person optimal. Therefore, the equations that we arrive in this section are necessary conditions for optimality and must be satisfied by any optimal threshold strategy.

First, we will provide a theorem, without including the proof here. The detailed proof can be found in Appendix A.

Theorem 1. *(a) Fix some $i \neq 0$ and suppose that γ_j has been fixed for all $j \neq i$. Then γ_i minimizes $J(\gamma)$ if and only if*

$$\gamma_i(Y_i) = \underset{d=1,\ldots,D}{\operatorname{argmin}} \sum_{j=1}^{M} \Pr(H_j|Y_i) \cdot a_i(H_j, d) \quad \text{with probability 1} \tag{2.9}$$

where

$$a_i(H_j, d) = E[C(\gamma_0(Y_0, U_1, \ldots, U_{i-1}, d, U_{i+1}, \ldots, U_N), U_1, \ldots, U_{i-1}, d, U_{i+1}, \ldots, U_N, H_j)|H_j] \tag{2.10}$$

(b) Suppose that the decision rules for peripheral nodes have been fixed, then γ_0 minimizes $J(\gamma)$ if and only if

$$\gamma_0(Y_0, U_1, \ldots, U_N) = \underset{d=1,\ldots,D}{\operatorname{argmin}} \sum_{j=1}^{M} \Pr(H_j|Y_0, U_1, \ldots, U_N) \cdot C(d, U_1, \ldots, U_N, H_j) \quad \text{with probability 1} \tag{2.11}$$

As we said before, the optimal strategy must be person-by-person optimal and thus if γ is an optimal strategy, then equations (2.9)-(2.11) hold. However, these equations do not lead to an optimal strategy. The reason is that the right hand side of (2.10) depends on the random variables $U_k, k \neq i$ (whose distribution depends on the coefficients $a_k(H_j, d)$). As it can be seen, the situation is very similar to the binary case we considered in section 2.1. At there, we had two likelihood ratio tests whose threshold depend on each other and in order to find the thresholds, we should solve coupled equations. Here, situation is somewhat similar; for any fixed choice of γ_0, we need to solve a system of $N \times M \times D$ nonlinear equations in as many unknowns.

Remark 1. If we want to minimize the probability of error (i.e. $C(u_0, u_1, \ldots, u_N, H_j)$ equals 0 when $u_0 = j$ and equals 1 when $u_0 \neq j$), then (2.11) simplifies to

$$\gamma_0(Y_0, U_1, \ldots, U_N) = \underset{d=1,\ldots,D}{\operatorname{argmax}} \ \Pr(H_d|Y_0, U_1, \ldots, U_N) \quad \text{with probability 1} \tag{2.12}$$

which is the classical maximum a posteriori probability (MAP) rule for hypotheses testing by the fusion center who has access to the observation Y_0 and the sensors messages $U_1, \ldots, U_N$.

Using Bayes rule for $i \neq 0$ we have:

$$\Pr(H_j|Y_i) = \frac{f(Y_i|H_j)\Pr(H_j)}{f(Y_i)} \tag{2.13}$$

replacing that in equations (2.9) and (2.10) and removing the constant term $f(Y_i)$, which does not depend on j:

$$\begin{aligned} \gamma_i(Y_i) &= \underset{d=1,\ldots,D}{\operatorname{argmin}} \sum_{j=1}^{M} \Pr(Y_i|H_j)\Pr(H_j)a_i(H_j, d) \\ &= \underset{d=1,\ldots,D}{\operatorname{argmin}} \sum_{j=1}^{M} \Pr(Y_i|H_j)b_i(H_j, d) \quad \text{with probability 1} \end{aligned} \tag{2.14}$$

where

$$b_i(H_j, d) = \Pr(H_j)a_i(H_j, d) \tag{2.15}$$

Also, for the fusion center, equation (2.10), we have:

$$\begin{aligned}\gamma_0(Y_0, U_1, \ldots, U_N) &= \underset{d=1,\ldots,D}{\operatorname{argmin}} \sum_{j=1}^{M} f(Y_0|H_j) \prod_{i=1}^{N} \Pr(\gamma_i(Y_i) = u_i|H_j)\Pr(H_j)C(d, U_1, \ldots, U_N, H_j) \\ &= \underset{d=1,\ldots,D}{\operatorname{argmin}} \sum_{j=1}^{M} f(Y_0|H_j)b_0(H_j, d, U_1, \ldots, U_N) \quad \text{w.p.1} \end{aligned} \tag{2.16}$$

where

$$b_0(H_j, d, U_1, \ldots, U_N) = \Pr(H_j)C(d, U_1, \ldots, U_N, H_j) \prod_{i=1}^{N} \Pr(\gamma_i(Y_i) = u_i|H_j) \tag{2.17}$$

Remark 2. (The shape of the decision regions at the peripheral sensors) Consider the M-dimensional space of all likelihood vectors of sensor $S_i, i \neq 0$. By the likelihood vector, we mean a vector whose j'th component is $f(Y_i|H_j)$, which from (2.14) we understand that it is the sufficient statistic for sensor S_i. The decision rule in (2.14) is equivalent to dividing this M-dimensional space into D regions, and deciding to send $u_i = d$ to the fusion center if the vector of likelihood belongs to the d'th region. Each region is specified by a set of linear equalities and therefore is a polyhedral. Note that this structure is the same as the structure of the optimal decision rule for the classical M-ary hypotheses testing.

Remark 3. (The shape of the decision regions at the fusion center) Similar to the previous remark, we can argue that the decision regions for the fusion center are M polyhedral regions in the M-dimensional space of all likelihood vectors of S_0. These

M regions are defined by a set of linear inequalities. However, the coefficients of these linear inequalities, $b_0(H_j, d, U_1, \ldots, U_N)$, depend on the vector of sent messages from the peripheral sensors, i.e. the realizations of $(U_1, \ldots, U_N)$. There are D^N realizations of these vectors, thus in the case that the fusion center doesn't have any observation of its own (no Y_0) the fusion rule is simply a deterministic function from $\{1, \ldots, D\}^N \to \{1, \ldots, M\}$.

2.3 Neyman-Pearson Framework

In this section we study the procedure for finding Neyman-Pearson optimum distributed detection. Throughout this section we do not have the simplifier assumption of conditionally independent observations. Also, we only consider the binary hypotheses testing, since the results for more general cases are extremely complicated. In this section we mainly focus on the work of Yan and Blum [3]. However, we may point out to some results from other papers as well.

2.3.1 Problem Definition

The problem setting is almost the same as what was mentioned in section 2.2.1. However, we have only two hypotheses (H_1 and H_2) here (thus $M = 2$), and also each sensor's message can be selected from an alphabet of size two (thus $D = 2$). Still we denote the observations by $Y_1, \ldots, Y_N$, and we assume that the fusion center does not have any observation. We let $\gamma_0(\mathbf{u})$ denote the probability that we decide on $U_0 = 1$ for a given set of sensors decisions $\mathbf{u} = (u_1, \ldots, u_N)$. We also let $\gamma_k(y_k)$ denote the probability that we decide on $U_k = 1$ for a given observation y_k.

In the Neyman-Pearson framework, we want to find a γ that maximizes the probability of detection which is defined as $P_d(\gamma) = \Pr(U_0 = 2|H_2)$ subject to the constraint $P_f(\gamma) =$

$\Pr(U_0 = 2|H_1) \leqslant \alpha$ where $0 \leqslant \alpha \leqslant 1$. P_f is called the probability of false alarm. It was shown in [2] that the form of the optimal distributed detection strategy under NP criterion is complicated, especially when the observations are not conditionally independent. However, the optimality of the monotone threshold strategies under certain assumptions was established there [2]. In the following, we will provide theorems giving conditions on the optimum strategy in the mentioned setting.

2.3.2 Finding The Optimum Strategy

To begin, let's define D_{jk}, where $j = 1, 2$ and $k = 1, \ldots, N$ as following:

$$D_{jk} = f(y_k|H_j) \sum_{\widetilde{u_k}} [\Pr(U_0 = 2|\widetilde{U_k} = \widetilde{u_k}, U_k = 2) - \Pr(U_0 = 2|\widetilde{U_k} = \widetilde{u_k}, U_k = 1)]$$
$$\times \Pr(\widetilde{U_k} = \widetilde{u_k}|Y_k = y_k, H_j) \qquad (2.18)$$

where

$$\widetilde{U_k} = (U_1, \ldots, U_{k-1}, U_{k+1}, \ldots, U_N)$$

and

$$\Pr(U_0 = 2|\widetilde{U_k} = \widetilde{u_k}, U_k = u_k) = \Pr(U_0 = 2|\underline{U} = \underline{u})$$

describes the decision rule at the fusion center. The sum in above, is over all possible values of $\widetilde{u_k}$. Now, we provide the following theorem without including the proof here. The detailed proof can be found in [3]. This problem is also studied in [2].

Theorem 2. *Given a fixed decision rule for fusion center and all the peripheral sensors but sensor k and also a set of observation distributions under H_1 and H_2 for all the peripheral sensors except the sensor k (i.e. $f(y_k|H_j), j = 1, 2$) such that $\frac{D_{2k}(Y_k)}{D_{1k}(Y_k)}$ and $f(y_k|H_j), j = 1, 2$*

have no point masses, we have:

1) *A* γ_k *of the form*

$$\gamma_k(y_k) = \begin{cases} 2 & \textit{if } D_{2k}(y_k) > \lambda_k D_{1k}(y_k) \\ 1 & \textit{if } D_{2k}(y_k) < \lambda_k D_{1k}(y_k) \end{cases}$$

will satisfy the Neyman-Pearson criterion for the given fusion rule and the given set of sensor's rules and also the conditional observation distributions. The event $D_{2k}(y_k) = \lambda_k D_{1k}(y_k)$ *which occurs with zero probability can be assigned to any hypotheses.*

2) *Any rule that satisfies the Neyman-Pearson criterion for the given fusion rule and the given set of sensor's rules and also the conditional observation distributions must be of this form except possibly on a set having zero probability under* H_1 *and* H_2*.*

This theorem gives the conditions for person-by-person optimality. This means that no better rule can be found by changing only one sensor at at time. The reason is that in above theorem we assumed that all the other decision rules are fixed and solved the optimization problem only over one sensor, so that is the mos optimal thing that we can achieve. What will happen if we change two sensors at a time? In the next theorem we consider this situation and show that by changing two sensors at a time, we can put more restrictions on the conditions provided by Theorem 2 such that $\lambda_1 = \lambda_2 = \ldots = \lambda_N \geqslant 0$ will produce an optimum solution.

Theorem 3. *Under the same assumptions as Theorem 2 and if the pdf of* $D_{2k}(Y_k)/D_{1k}(Y_k)$ *under* $H_j, j = 1, 2$ *is positive for* $0 < D_{2k}(Y_k)/D_{1k}(Y_k) < \infty$*, the optimal strategy have a set of sensor rules* $\gamma_1, \ldots, \gamma_N$ *that have the conditions described in Theorem 2 as well as the new condition:* $\lambda_1 = \lambda_2 = \ldots = \lambda_N = \lambda$*. Thus, only a set of sensor rules* $\gamma_1, \ldots, \gamma_N$ *of* *the form*

$$\gamma_k(y_k) = \begin{cases} 2 & \textit{if } D_{2k}(y_k) > \lambda D_{1k}(y_k) \\ 1 & \textit{if } D_{2k}(y_k) < \lambda D_{1k}(y_k) \end{cases}$$

will satisfy the Neyman-Pearson criterion for a given fusion decision rule. The event $D_{2k}(y_k) = \lambda D_{1k}(y_k)$ *which occurs with zero probability can be assigned to any hypotheses.*

The proof is complex and contains massive integral calculations. So, we do not include it here, but a rigorous proof can be found in [3].

In the above results, we did not assume anything about the dependency of the conditional observations. However, it is shown in [2] that under the assumption that observations are conditionally independent, if there exists an optimal strategy, then there also exists a monotone threshold strategy that is optimal.

As we can see, although in the centralized detection, the Bayesian and Neyman-Pearson problems are almost equivalent (with the suitable choice of costs of course), the situation is somewhat more complex in the decentralized detection scheme.

Chapter 3

Architecture and Channel Considerations of Wireless Sensor Networks

The classical framework does not adequately take into account important features of sensor network architecture and of the wireless channel. In this chapter, we will review some works that consider these features [4, 5, 6]. First we will review the situations where we have some restrictions on the wireless channel between the peripheral sensors and the fusion center. The main paper that we review here is the work of J.F. Chamberland and V. Veeravalli [4], who consider the scenario where the sensor network is constrained by the capacity of the wireless channel over which the sensors and the fusion center communicate. Also, we will review the scenario where we have a broadcast channel (and no central node) and all the nodes try to reach a consensus by successive retesting and rebroadcasting of the updated decision [5].

A common architecture for wireless sensor networks is the tree configuration. Here, we will briefly the results of [6] which reviewed the optimal strategies for sensor nodes (i.e. local functions, which are based on the messages received from predecessors and own observations) and also the decision rule at fusion in terms of optimizing the error exponent.

3.1 Wireless Channel Considerations

3.1.1 Multiple Access Channel

In previous chapter, we studied the case where each of the N sensors chooses its message from the set $\{1, \ldots, D\}$. Therefore, the quantity of information that is sent through the channel to the fusion center does not exceed $N\lceil \log_2 D \rceil$. However in real-world wireless sensor networks the multiple access channel available to the sensors is limited. In these cases, we should take into account this new constraint in the design problem. Here, we assume that the sensor S_i can send its message from a set of size D_i, and thus we need to have:

$$\sum_{i=1}^{N} \lceil \log_2(D_i) \rceil \leqslant R \tag{3.1}$$

where R is the capacity of the channel. In general, the admissible rate of a practical system with a simple encoding scheme may depend on bandwidth, power, noise density, and maximum tolerable bit error rate at the output of the decoder. However, we disregard the specific values of these parameters. Also, we neglect communication errors and assume that we can transmit reliably at a maximum rate of R bits per unit time. We call a strategy admissible if it satisfies (3.1). The setting of the problem is illustrated in the Fig. 3.1. Also, we assume that the observations are conditionally independent.

Here, we have slightly different notations with respect to the previous chapter. Besides the fact that each sensor has its own size of alphabet ($\{D_1, \ldots, D_N\}$), we also assume that

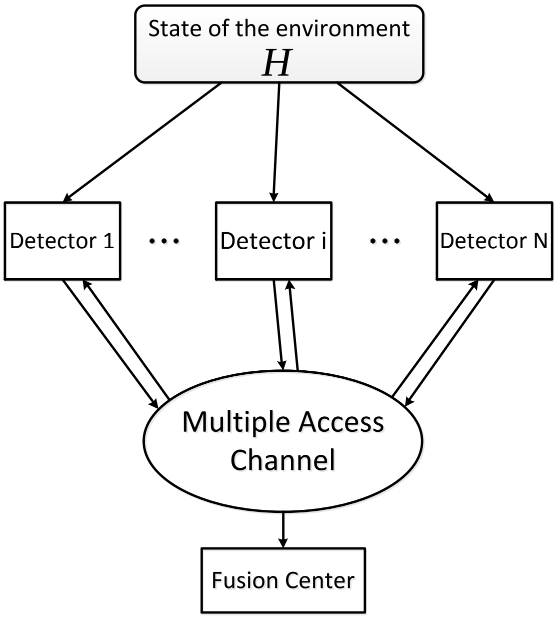

Fig. 3.1 A wireless sensor network where sensors transmit their messages to the fusion center through a multiple access channel

sensors take observations over time with each sensor receiving a sequence of observations ($\{Y_{i,t} : t = 1, 2, \ldots, T\}$). In the asymptotic regime where the observation interval goes to infinity ($T \to \infty$), we can define the error exponent measure as follows:

$$C(\gamma) = -\lim_{T\to\infty} \frac{1}{T} \log P_e^{(T)}(\gamma) \tag{3.2}$$

where $P_e^{(T)}(\gamma)$ is the probability of error at the fusion center when a maximum a posteriori detector is used. Note that because for any reasonable transmission strategy, the associated probability of error at the fusion center goes to zero exponentially fast as grows unbounded [4], the error exponent is a suitable way to compare transmission schemes. However, since C is a monotonic decreasing function of $P_e^{(T)}(\gamma)$, comparing C or $P_e^{(T)}(\gamma)$ to compare strategies is exactly the same (except for the change of inequality direction).

The error exponent is also known as the Chernoff information.

For a multiple access channel that is able to carry R bits of information per unit time, we can pose the design problem formally as finding an admissible strategy that maximizes the error exponent defined in (3.2).

As we saw in chapter 2, using MAP detection in fusion center minimizes the probability of error. So we use MAP detection as fusion rule and try to evaluate the error exponent for an admissible strategy γ using Chernoff's theorem.

Theorem 4. (Chernoff) *Suppose γ is an admissible strategy. The best achievable exponent in the probability of error at the fusion center is given by:*

$$C(\gamma) = -\min_{0 \leqslant s \leqslant 1} \log \left[\sum_{\underline{u} \in \Upsilon} (\Pr(\underline{u}|H_1))^s (\Pr(\underline{u}|H_2))^{1-s} \right] \tag{3.3}$$

where

$$\Upsilon = \{1, \ldots, D_1\} \times \ldots \times \{1, \ldots, D_N\}$$

The proof of this theorem can be found in information theory books (see e.g., [7]). Using Chernoff's theorem, the design problem is now changed to finding an admissible strategy that maximizes the Chernoff information defined in (3.3).

Now we provide two propositions from [4] (without proofs) that derive a set of conditions that simplify the design problem greatly.

Proposition 1. For strategy γ, the contribution $C_{S_i}(\gamma)$ of sensor S_i to the Chernoff information is upper bounded by the Chernoff information contained in one observation Y:

$$C_{S_i}(\gamma) \leqslant C^* \triangleq -\min_{0 \leqslant s \leqslant 1} \log \left[\int_{\mathcal{Y}} (\Pr(y|H_1))^s (\Pr(y|H_2))^{1-s} dy \right] \tag{3.4}$$

In words, this proposition means that the contribution of a single sensor to the total

Chernoff information cannot exceed the information contained in each observation. Using this proposition, we will find a sufficient condition for which having R binary sensors is optimal. To do so, define $C_1(\gamma_i)$ to be the Chernoff information corresponding to a single sensor with decision rule γ_i:

$$C_1(\gamma_i) = -\min_{0 \leqslant s \leqslant 1} \log \left[\sum_{u=1}^{D_i} (\Pr(\gamma_i(y) = u | H_1))^s (\Pr(\gamma_i(y) = u | H_2))^{1-s} \right] \tag{3.5}$$

Proposition 2. Assume that $\widetilde{\gamma}_b$ is a binary strategy for one of the peripheral sensors (i.e. based on its observation, the sensor transmits one of its two possible messages to the fusion center). If the condition

$$C_1(\widetilde{\gamma}_b) \geqslant \frac{C^*}{2} \tag{3.6}$$

holds, then having R identical sensors, each sending one bit of information (i.e. binary decision rule), is optimal.

In summary, when there is a limit, R, on the capacity of the multiple access channel between peripheral sensors and the fusion center, then under the condition (3.6), the optimal strategy consists of R identical sensors each utilizing binary decision rules (same as the likelihood ratio decision rules introduced in chapter 2). Two special cases of Gaussian and Exponential observations has been studied in [4], and it was shown that having identical binary sensors is optimal in the asymptotic regime where the observation interval goes to infinity. In other words, the gain offered by having more sensors outperforms the benefits of getting detailed information from each sensor whenever the number of observations per sensor is large. However, it was also shown through counterexamples that choosing binary decision rules is not always the optimum strategy. Moreover, having identical binary sensors may not be optimal when observations are dependent across sensors. The reason is that when the observations are dependent under the same hypothesis across sensors having fewer sensors sending multiple bits may result in better performance.

3.1.2 Broadcast Channel

Until now, we have focused on detection systems with “parallel” or “star” topology, which means that each of the peripheral sensors transmit its message directly to the fusion center. More over, although we called the detection decentralized, in all the schemes we saw, there is a central node that receives all the messages from sensors and then decides on the true hypothesis. In this section we explore an alternative approach, which by using feedback (of all sensor decisions to all sensors) and successive retesting and rebroadcasting (which is called *parley* operation), tries to reach a common decision or consensus among nodes. In this setting (see Fig. 3.2) there is no central node and the fully connected sensors communicate through a broadcast channel. Here, we review the work of P.F.Swaszek and P.Willett [5]. The algorithm they use to reach a consensus is as follows:

1- Each sensor receives an observation and based on that, computes its likelihood ratio observation.

2- The sensors make initial decisions based on their observations and broadcast their decisions to all the other sensors.

3- Based on the original observation and also the decisions of all other sensors, each sensor revises its previous decision, and sends it to all other nodes again.

4- Step 3, which is called parleying is repeated until a consensus is reached. A consensus here means that all sensors agree.

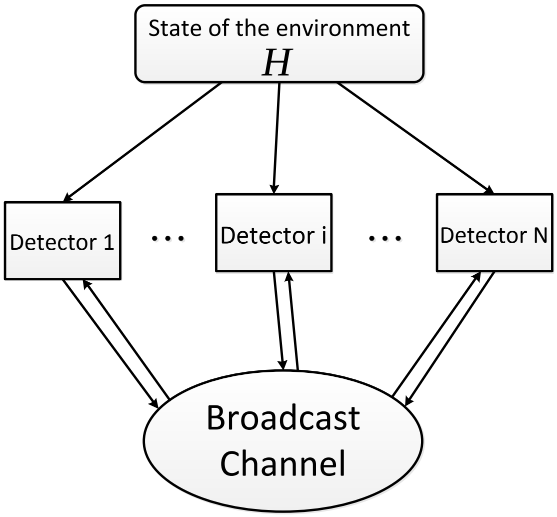

Fig. 3.2 Fully-connecetd, yet decentralized detection network

It is clear that we like a consensus to be both "fast" and "correct". It is mentioned in [5] that experience has shown that a general optimization of both is infeasible. So, they proposed two algorithms that each tries to optimize one of them; a greedy approach which convergence is quite fast but the correctness is sacrificed and the N-th root approach which is delayed somewhat but the decision is the best possible, given data. We will review them briefly below.

Greedy Approach

We assume that we have a binary hypotheses testing problem (H_1 and H_2). Also, we assume that the observations are conditionally independent. Let us denote the likelihood ratio for sensor i as:

$$\Lambda(y_i) = \frac{f(y_i|H_2)}{f(y_i|H_1)} \tag{3.7}$$

Also, denote the decision of the ith sensor at stage m as $u_{i,m} \in \{1, 2\}$. Note that at stage $m + 1$, the ith sensor uses the decisions of all other sensors up to stage m and its own observation. Thus, if we assume the optimality criterion to be the probability of error, the optimal Bayesian decision of the node i at stage m is:

$$\Lambda(y_i) \underset{H_1}{\overset{H_2}{\gtrless}} \frac{\Pr(H_1) \cdot \Pr(\mathcal{U}_{i,m}|H_1)}{\Pr(H_2) \cdot \Pr(\mathcal{U}_{i,m}|H_2)} \tag{3.8}$$

where $\mathcal{U}_{i,m} = \{u_{k,n} : k \neq i, n = 1, \ldots, m-1\}$ denotes the set of all decisions of sensors (except sensor i) until stage $m-1$. The proof is not provided in [5], but we include it here. For the Bayesian test with uniform costs we have:

$$\Pr(H_2|y_i,\mathcal{U}_{i,m}) \underset{H_1}{\overset{H_2}{\gtrless}} \Pr(H_2|y_i,\mathcal{U}_{i,m}) \tag{3.9}$$

$$\Rightarrow \Pr(H_2) \cdot \Pr(y_i,\mathcal{U}_{i,m}|H_2) \underset{H_1}{\overset{H_2}{\gtrless}} \Pr(H_1) \cdot \Pr(y_i,\mathcal{U}_{i,m}|H_1) \tag{3.10}$$

$$\Rightarrow \Pr(H_2) \cdot \Pr(y_i|H_2)\Pr(\mathcal{U}_{i,m}|H_2) \underset{H_1}{\overset{H_2}{\gtrless}} \Pr(H_2) \cdot \Pr(y_i|H_1)\Pr(\mathcal{U}_{i,m}|H_1) \tag{3.11}$$

$$\Rightarrow \Lambda(y_i) = \frac{f(y_i|H_2)}{f(y_i|H_1)} \underset{H_1}{\overset{H_2}{\gtrless}} \frac{\Pr(H_1) \cdot \Pr(\mathcal{U}_{i,m}|H_1)}{\Pr(H_2) \cdot \Pr(\mathcal{U}_{i,m}|H_2)} \tag{3.12}$$

where in deriving (3.11) we used the fact that the decisions of other sensors are independent from the observation y_i of i-th sensor.

So, in each stage, the nodes update their thresholds, using (3.8), which is the locally optimal decision (but not globally). Through a couple of propositions, it is shown in [5] that such an algorithm will reach a consensus with probability one. Also through some simulations, it is shown that although the convergence is quite fast, the correctness is sacrificed.

The N-th Root Approach

In the previous method, the decisions of sensors were locally optimal. This was the reason of the probable convergence to the wrong hypothesis. However, in this pert, we explore an algorithm which is globally optimal and its solution is exactly the same as the centralized scheme. The cost that we pay to achieve this precision is the delay of the algorithm in converging to the final solution.

We know that assuming either a Bayes or NP performance criterion the optimum centralized test for binary hypotheses binary testing is:

$$\prod_{i=1}^{N} \Lambda(y_i) \underset{H_1}{\overset{H_2}{\gtrless}} \lambda \tag{3.13}$$

Now, assuming that the i-th sensor performs a test like

$$\Lambda(y_i) \underset{H_1}{\overset{H_2}{\gtrless}} \lambda_{i,m} \tag{3.14}$$

at stage m, global convergence means either $\Lambda(y_i) > \prod_{i=1}^{N} \lambda_{i,m}$ or $\Lambda(y_i) < \prod_{i=1}^{N} \lambda_{i,m}$. In the previous part, the thresholds were posterior likelihood ratios. However, here, we need to choose thresholds such that $\prod_{i=1}^{N} \lambda_{i,m}$ converges to λ. If this condition holds, then we can make sure that if a consensus decision is reached, then it would match the centralized decision exactly. It was shown in [5] that for the following initial values and update rules of the thresholds, a consensus will be reached with probability one:

$$\text{Initial Values} \; : \; \lambda_{i,1} = \lambda^{1/n} \tag{3.15}$$

$$\text{Update rule} \; : \; \lambda_{i,m} = \lambda_{i,m-1} \cdot \frac{\Pr(\mathcal{U}_{i,m}|H_1)}{\Pr(\mathcal{U}_{i,m}|H_2)} \tag{3.16}$$

As we said before, this approach will find the same solution as the centralized scheme, and thus has better performance with respect to greedy algorithm. However, the consensus does not happen as fast as the greedy algorithm.

3.2 Tree Network Architecture

In all previous parts (except the broadcast channel section), we assumed that the network's architecture is star, that every nodes transmit directly to the fusion center. In this section we consider the problem of decentralized detection in a network consisting of a large number of nodes arranged as a tree of bounded height (the number of hops needed to reach the fusion center from a node). Tree is one of the most common architectures for wireless sensor networks. We briefly only review the results of [6] here.

In the tree configuration, we have a directed graph, with the fusion center as the root. First, the leaves send their messages (which are computed based on nodes' observations) to their ancestors. From then, based on its observation and also the received messages of all its predecessors, a node compute its message and sends it to its ancestor and so on. The same procedure goes on until the fusion center receives the messages of all its predecessors. Then, it decides about the true hypothesis. It is shown in [6] that under Neyman-Pearson formulation and certain assumptions (like independent observations, etc.), the network can achieve the same performance as if all nodes were transmitting directly to the fusion center. Also, they show that the probability of false alarm decays exponentially fast with the number of nodes.

The tree configuration is also studied in [8] where a numerical algorithm is developed for determining the optimal decision rule at each node under Bayesian formulation.

Chapter 4

Gossip-Based Decentralized Detection

In this chapter, we are going to propose a gossip-based fully decentralized detection algorithm. To the best of our knowledge, the proposed methods is novel. The algorithms that we studied until now, were mostly semi-decentralized, in the sense that all the peripheral sensors transmit a quantized function of their observations to a central node (fusion center). So, it is clear that we still have the problems of centralized systems such as single point of failure, data management (flooding more data to fusion center than it can process), and security. The only truly decentralized algorithm we reviewed here is the one studied in section 3.1.2 which utilizes a broadcast channel to reach a consensus among all sensors, without using any centralized processing. However, the assumption that all the nodes can communicate with each other through a broadcast channel is not realistic. In practical wireless sensor networks, because of the power constraints, nodes are only capable of short-range communications. Thus, each node can communicate with only a few other nodes that are close to it.

Our proposed gossip-based algorithm, tries to reach a consensus among all the nodes by only *local communications*. We will propose our method in the first section and show in the following sections that it converges and the solution is exactly the same as the solution of the centralized scheme, and hence globally optimum.

4.1 Gossip-Based Algortihm

Assume that we have a wireless sensor nodes, with N sensors, and without any central node. Also, assume that we have a binary hypotheses testing problem. Sensor S_i can communicates only with a few other nodes that are in its communication range. We call these nodes the neighbors of sensor S_i and denote them by the set V_i. The notation introduced in section 2.2.1 still holds. Our proposed algorithm for fully decentralized detection is briefly as follows. Full details can be found in the next section.

1- Each sensor receives an observation and based on that, computes its likelihood ratio. Let us denote this likelihood ratio by $\Lambda^{(0)}$:

$$\Lambda_i^{(0)} = \frac{f(y_i|H_2)}{f(y_i|H_1)}, \qquad i = 1, \dots, N \tag{4.1}$$

2- The sensors make initial decisions based on their observations. Note that they make their decisions under either Bayesian or Neyman-Pearson criterion. Note that, since we have a binary hypothesis testing, under either of the frameworks, all the nodes implement a likelihood ratio test.

3- **Gossiping:** After calculating the likelihood ratios and making initial decisions, the nodes perform several rounds of gossiping. In each round, the nodes communicate two-by-two and update their likelihood ratios. If the nodes i and j communicate with each other (or "gossip") at iteration k, then, the update procedure is as follows:

$$\Lambda_i^{(k)} = \sqrt{\Lambda_i^{(k-1)} \cdot \Lambda_j^{(k-1)}} \tag{4.2}$$

$$\Lambda_j^{(k)} = \sqrt{\Lambda_i^{(k-1)} \cdot \Lambda_j^{(k-1)}} \tag{4.3}$$

where $\Lambda_i^{(k)}$ denotes the value of the likelihood ratio of sensor S_i at iteration k of the algorithm $(0 \leqslant k)$.

4- Based on their new likelihood ratios, the sensors update their decisions. As mentioned in step 2, all the nodes implement a likelihood ratio test with fixed thresholds. Unlike the work of [5], in our algorithm, the likelihood ratios are updated and the thresholds of the test are fixed and the same for all the sensors ($\lambda^{1/N}$). More details are provided in section 4.2.

5- Steps 3 and 4 are repeated until all the sensors decide on the same hypothesis (denoted as "convergence").

In next section we will show that the mentioned algorithm will converge to the same decision as the centralized scheme.

4.2 Analysis of Proposed Algorithm

Having N conditionally independent observations $(Y_1, \ldots, Y_N)$, and assuming a Bayes cost or Neyman-Pearson performance criterion, the optimum (centralized) test for the binary hypotheses testing is well known to be:

$$\prod_{i=1}^{N} \Lambda(y_i) \underset{H_1}{\overset{H_2}{\gtrless}} \lambda \tag{4.4}$$

where in NP framework, λ can be found based on the probability of error criterion, and in Bayesian framework it can be found based on the prior probabilities of the two hypotheses and the costs. By optimum decision, we mean a decision which minimizes the probability of the error.

In a fully decentralized sensor network, each node is only aware of its own likelihood ratio and can perform a test like:

$$\Lambda_i(y_i) \underset{H_1}{\overset{H_2}{\gtrless}} \lambda_i \tag{4.5}$$

In our method we have equal thresholds for all the nodes, and choose them in a way such that their product is equal to the appropriate λ in (4.4). Therefore, $\lambda_i = \lambda^{1/N}, \quad i = 1, \ldots, N$. These thresholds remain fixed through the algorithm. However, each node updates its likelihood ratio in each iteration of the algorithm. The procedure is as follows: in each iteration, nodes gossip two-by-two with each other and update their likelihood ratios as described in (4.2) and (4.3). Note that although the individual likelihood ratios change, the product of the likelihood ratios of all nodes remain fixed through the algorithm. In other words:

$$\prod_{i=1}^{N} \Lambda_i^{(k)} = \prod_{i=1}^{N} \Lambda_i^{(k-1)}, \quad 1 \leqslant k \tag{4.6}$$

It is clear that only $\Lambda_i^{(0)}$ has the classical definition of the likelihood ratio for sensor S_i, and hence equals to $\frac{f(y_i|H_2)}{f(y_i|H_1)}$. However, we still call the updated versions, $\Lambda_i^{(k)}$for $1 \leqslant k$, likelihood ratios for ease of notation.

Then, based on the updated versions of likelihood ratios, the nodes revise their decisions. The algorithm will converge if all the nodes have the same decisions. Assume that at stage k all the nodes have the same decisions. Without loss of generality, assume that they all decide on H_2. Thus:

$$\Lambda_i^{(k)} > \lambda_i = \lambda^{1/N}, \quad i = 1, \ldots, N \tag{4.7}$$

Multiplying all the inequalities, we have:

$$\begin{aligned}&\prod_{i=1}^{N}\Lambda_i^{(k)} > \lambda\\ \Rightarrow &\prod_{i=1}^{N}\Lambda_i^{(0)} > \lambda \qquad \text{(using (4.6))}\end{aligned}$$

Which shows that H_2 is also the solution of the centralized test, and hence optimum (minimizes the probability of error). Thus, if all the nodes "agree" with each other and reach a consensus decision, then their decision matches the centralized decision exactly. However, a big question remains; Does this algorithm converge or not?

Note that, in order to prove that the mentioned algorithm converges, it is enough to show that

$$\text{as}\ \ k \to \infty: \quad \Lambda_i^{(k)} \to \left[\prod_{j=1}^{N}\Lambda_j^{(0)}\right]^{1/N}, \quad i = 1, \ldots, N \tag{4.8}$$

The reason is that in this case, all the likelihood ratios become equal. Also, we know that the thresholds are equal by definition. Therefore, all the nodes perform the same likelihood ratio test and thus get the same result (convergence). The type of convergence that we are concerned about in (4.8), is convergence in expectation. In the following theorem, we prove that (4.8) is true and hence, our proposed algorithm converges.

Theorem 5. *Assume that we have N nodes with initial values $\Lambda_i^{(0)},\ \ i = 1, \ldots, N$. Consider the following gossip algorithm: At each iteration, nodes choose one of their neighbors at random and gossip with each other. When two nodes gossip with each other, they update their values according to the following procedure:*

$$\Lambda_i^{(k)} = \sqrt{\Lambda_i^{(k-1)} \cdot \Lambda_j^{(k-1)}} \tag{4.9}$$

$$\Lambda_j^{(k)} = \sqrt{\Lambda_i^{(k-1)} \cdot \Lambda_j^{(k-1)}} \tag{4.10}$$

where $\Lambda_i^{(k)}$ denotes the value of the likelihood ratio of sensor S_i at iteration k of the algorithm $(0 \leqslant k)$. This gossip algorithm converges in expectation and we have:

$$\lim_{k \to \infty} E\left[\Lambda_i^{(k)}\right] = \left[\prod_{j=1}^{N} \Lambda_j^{(0)}\right]^{1/N}, \quad i = 1, \ldots, N \tag{4.11}$$

The proof can be found in Appendix B.

Another important question is: How fast does the algorithm converge? *or* How many messages do we need to send among sensors in order to converge? This can highly affect the energy consumption of the sensors (the more transmitted messages, the more battery consumption), and also the delay of the system. Actually there are many fast gossip algorithms that we can employ. For example, the algorithm proposed in [9], achieves ϵ-accuracy with high probability after $O\big(n \log\log n \log \frac{kn}{\epsilon}\big)$ messages.

Discussion

Although the idea of gossiping for decentralized failure detection [10], threshold detection [11], and successive retesting and rebroadcasting [5] have been studied before, our proposed algorithm is hugely different from them in the following aspects:

- As mentioned before, the likelihood ratios are updated in our algorithm in each iteration, while other algorithms mainly try to update their thresholds.

- Unlike [5] we do not assume that all the nodes can communicate with each other through a broadcast channel, and only consider local communications among nodes (more realistic in the sense of power constraints).

- Our algorithm reaches a consensus decision for sure, and its decision matches the centralized decision exactly, and hence optimum in the sense of probability of error.

Appendix A

Proof of Theorem 1

This section is devoted to the proof of Theorem 1. The proof is taken from [2].

Lemma A.1. Let Z be a random variable taking values in $\mathcal{Z}$. Also assume that conditioned on any hypothesis, Z is independent of X. Let $F : \{1, \ldots, D\} \times \mathcal{Z} \times \{H_1, \ldots, H_M\} \rightarrow \mathbb{R}$ be a given cost function. Let δ^* be an element of Δ. Then, δ^* minimizes $E[F(\delta(x), Z, H)]$ over all $\delta \in \Delta$ if and only if

$$\delta^*(X) = \underset{d=1,\ldots,D}{\operatorname{argmin}} \sum_{j=1}^{M} a(H_j, d)\Pr(H_j|X), \quad \text{with probability 1}$$

where

$$a(H_j, d) = E[F(d, Z, H_j)|H_j] \quad \forall j, d$$

Proof. The minimization of $E[F(\delta(x), Z, H)]$ over all $\delta \in \Delta$ is equivalent to finding a $\delta(X)$ that minimizes $E[F(d, Z, H)|X]$, over all $d \in \{1, \ldots, D\}$, with probability 1. Using the properties of conditional expectations, we have:

$$\begin{aligned} E[F(d,Z,H)|X] &= E[E[F(d,Z,H)|H,X]|X] \\ &= E[E[F(d,Z,H)|H]|X] \qquad\qquad \text{(independence of } X \text{ and } Z\text{)} \\ &= \sum_{j=1}^{M} E[F(d,Z,H_j)|H_j]\Pr(H_j|X) \end{aligned}$$

which is exactly what we wanted to achieve.

Using the above lemma, the proof is straight forward. To prove part (a) of theorem 1, notice that we are concerned with the minimization of

$$E[C(\gamma_0(Y_0,U_1,\ldots,U_{i-1},\gamma_i(Y_i),U_{i+1},\ldots,U_N),U_1,\ldots,U_{i-1},\gamma_i(Y_i),U_{i+1},\ldots,U_N,H_j)]$$

with respect to γ_i. This is of the form considered in Lemma A.1, provided that we identify X with Y_i, Z with $(Y_0,U_1,\ldots,U_{i-1},U_{i+1},\ldots,U_N)$, and $F(d,Z,H_j)$ with $C(\gamma_0(Y_0,U_1,\ldots,U_{i-1},d,U_{i+1},\ldots,U_N),U_1,\ldots,U_{i-1},d,U_{i+1},\ldots,U_N,H_j)$. The result then follows from Lemma 1.

To prove part (b), note that:

$$\begin{aligned} \min_{\gamma_0} J(Y_0,\gamma_1,\ldots,\gamma_N) &= E\left[\min_{d=1,\ldots,M} E[C(d,U_1,\ldots,U_N,H)|Y_0,U_1,\ldots,U_N]\right] \\ &= E\left[\min_{d=1,\ldots,M} \sum_{j=1}^{M} \Pr(H_j|Y_0,U_1,\ldots,U_N)C(d,U_1,\ldots,U_N,H_j)\right] \end{aligned}$$

Q.E.D.

Appendix B

Proof of Theorem 5

We use the result of [12] to prove Theorem 5. It was shown in [12] that:

Lemma B.1. Assume that we have N nodes with initial values $x_i^{(0)}, \quad i = 1, \ldots, N$. Consider the following gossip algorithm: At each iteration, nodes choose one of their neighbors at random and gossip with each other. When two nodes (i and j) gossip with each other, they update their values according to the following procedure:

$$x_i^{(k)} = \frac{x_i^{(k-1)} + x_j^{(k-1)}}{2} \tag{B.1}$$

$$x_j^{(k)} = \frac{x_i^{(k-1)} + x_j^{(k-1)}}{2} \tag{B.2}$$

where $x_i^{(k)}$ denotes the value of the sensor i at iteration k of the algorithm $(0 \leqslant k)$. This gossip algorithm converges in expectation and we have:

$$\lim_{k\to\infty} E\left[x_i^{(k)}\right] = \frac{\left[\sum_{j=1}^{N} x_j^{(0)}\right]}{N}, \quad i = 1, \ldots, N \tag{B.3}$$

The proof of the lemma can be found in [12]. Now, if we do the following replacement:

$$x_i^{(k)} \triangleq \log(\Lambda_i^{(k)}) \quad \text{for} i = 1, \ldots, N \quad \text{and} \quad 0 \leqslant k \tag{B.4}$$

then Lemma B.1 transforms to Theorem 5, because the update procedure in Theorem 5:

$$\Lambda_i^{(k)} = \sqrt{\Lambda_i^{(k-1)} \cdot \Lambda_j^{(k-1)}} \tag{B.5}$$

is equivalent to

$$\log\left(\Lambda_i^{(k)}\right) = \frac{\log\left(\Lambda_i^{(k-1)}\right) + \log\left(\Lambda_j^{(k-1)}\right)}{2} \tag{B.6}$$

which is the update rule of Lemma B.1. Also, we have:

$$\log\left(\left[\prod_{j=1}^{N} \Lambda_j^{(0)}\right]^{1/N}\right) = \frac{\sum_{j=1}^{N} \log\left(\Lambda_i^{(0)}\right)}{N} \tag{B.7}$$

which shows that Theorem 5 is equivalent to Lemma B.1 in the log domain and thus the proof is complete.

Q.E.D.

References

[1] R. Tenney and N. Sandell, “Detection with distributed sensors,” *Aerospace and Electronic Systems, IEEE Transactions on*, vol. AES-17, no. 4, pp. 501 –510, 1981.

[2] J. N. Tsitsiklis, “Decentralized Detection,” *Advances in Signal Processing*, vol. 2, pp. 297–344, 1993.

[3] Q. Yan and R. Blum, “Distributed signal detection under the Neyman-Pearson criterion,” *Information Theory, IEEE Transactions on*, vol. 47, pp. 1368 –1377, May 2001.

[4] J.-F. Chamberland and V. Veeravalli, “Decentralized detection in sensor networks,” *Signal Processing, IEEE Transactions on*, vol. 51, pp. 407 – 416, Feb. 2003.

[5] P. Swaszek and P. Willett, “Parley as an approach to distributed detection,” *Aerospace and Electronic Systems, IEEE Transactions on*, vol. 31, no. 1, pp. 447 –457, 1995.

[6] W. P. Tay, J. Tsitsiklis, and M. Win, “Data Fusion Trees for Detection: Does Architecture Matter?,” *Information Theory, IEEE Transactions on*, vol. 54, no. 9, pp. 4155 –4168, 2008.

[7] T. M. Cover and J. A. Thomas, *Elements of information theory.* New York, NY, USA: Wiley-Interscience, 1991.

[8] Z.-B. Tang, K. Pattipati, and D. Kleinman, “Optimization of detection networks. II. Tree structures,” *Systems, Man and Cybernetics, IEEE Transactions on*, vol. 23, no. 1, pp. 211 –221, 1993.

[9] K. Tsianos and M. Rabbat, “Fast decentralized averaging via multi-scale gossip,” *Proc. IEEE Distributed Computing in Sensor Systems*, June 2010.

[10] S. Ranganathan, A. D. George, R. W. Todd, and M. C. Chidester, “Gossip-style failure detection and distributed consensus for scalable heterogeneous clusters,” *Cluster Computing*, vol. 4, pp. 197–209, 2001.

[11] F. Wuhib, R. Stadler, and M. Dam, "Gossiping for threshold detection," in *Proceedings of the 11th IFIP/IEEE international conference on Symposium on Integrated Network Management*, IM'09, (Piscataway, NJ, USA), pp. 259–266, IEEE Press, 2009.

[12] S. Boyd, A. Ghosh, B. Prabhakar, and D. Shah, "Randomized gossip algorithms," *IEEE/ACM Trans. Netw.*, vol. 14, pp. 2508–2530, June 2006.